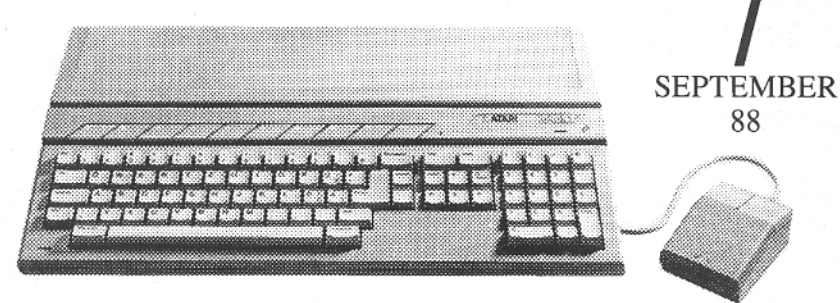


DM 4.50 SFR 6.00 ÖS 55

7

SEPTEMBER 88



DAS ATARI MAGAZIN VON USERN FÜR USER

USA-News
Assembler
Omikron Basic
Midi
BTX



DIE SPITZEN-BÜCHER ZUM NEUEN ATARI-BASIC



Heim Verlag

ATARI ST

Schneider/Steinmeier

kurz & klar

OMIKRON BASIC

Ein fundiertes Nachschlagewerk

Heim-Verlag



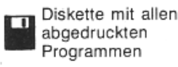
MERKMALE:

▶ BASIC-Einsteiger und Profis erfahren in diesem Buch, wie man professionelle Programme in Omikron-BASIC erstellt. ▶ Neben vielen ausführlich dokumentierten Listings, die zeigen, wie man in Omikron-BASIC optimal programmiert, finden Sie zahlreiche hilfreiche Tips und Anregungen, die Sie bei der Arbeit an Ihren eigenen Projekten verwenden können. ▶ Ein umfangreiches Kapitel beschäftigt sich ausschließlich mit dem Einsatz von GEM-Funktionen in Omikron-BASIC-Programmen. Hier finden Sie Erläuterungen zur Verwendung von Fenstern, Dialogboxen und Menüzeilen. Weiterhin wird eine auf der von Omikron mitgelieferten GEMLIB-Bibliothek aufbauende Library zur Programmierung der erwähnten GEM-Funktionen vorgestellt und erklärt. ▶ Die abgedruckten Listings wurden aus allen Anwendungsbereichen ausgewählt und zeigen, daß man Omikron-BASIC zur Erstellung von Utilities, Anwendungs-, Mathematik- und Grafikprogrammen benutzen kann. Natürlich finden Sie auch einige Spielprogramme in diesem Buch. ▶ Zum Buch gibt es eine Programmdiskette mit allen aufgeführten Übungs- und Beispielprogrammen. ▶ Ausführliche Übersicht der OMIKRON-BASIC-Befehle ▶ Über 50 Programme

INHALT:

▶ Tips und Tricks zur Programmierung ▶ GEM-Programmierung ▶ Utilities ▶ Grafik-Programmierung ▶ Anwendungsprogramme ▶ Mathematikprogramme ▶ Spiele in Omikron-BASIC

ca. 350 Seiten **DM 49,-**



Diskette mit allen abgedruckten Programmen **DM 39,-**

INHALT:

OMIKRON BASIC ist eine der leistungsfähigsten Programmier-Sprachen für den ATARI ST.

Um den komplexen Befehlssatz und die zahlreichen Möglichkeiten richtig einsetzen zu können, ist ein fundiertes Nachschlagewerk unbedingt erforderlich, damit das ständige Blättern in der Anleitung, in Zeitschriften und Büchern überflüssig wird.

In KURZ & KLAR finden Sie alle Befehle und deren Parameter in einer ausführlichen alphabetischen Übersicht.

Auch ein nach Sachgebieten geordnetes Verzeichnis der Anweisungen ist enthalten. Weiterhin erleichtern Tabellen und Übersichten die tägliche Arbeit.

Für die Besitzer des OMIKRON-BASIC-COMPILERS ist ein Abschnitt mit Hinweisen und Informationen über die Anpassung von Programmen des Interpreters gedacht.

Über 200 Seiten **DM 29,-**

MERKMALE:

OMIKRON-BASIC stellt eine der leistungsfähigsten Programmiersprachen für den ATARI ST dar. Trotz seines großen Umfangs ist es gerade auch für den Programmier-Anfänger ideal geeignet.

Das jetzt vorliegende GROSSE OMIKRON-BASIC-BUCH bietet dem Anfänger einen hervorragenden Einstieg in diese komfortable Programmiersprache. Das Buch ist leicht verständlich geschrieben, sodaß der Neuling ohne Probleme seine eigenen Programme erstellen kann. Aber auch für den fortgeschrittenen Anwender und Aufsteiger ist dieses Buch die richtige Grundlage.

Der 1. Hauptteil des Buches ist eine systematische Einführung in die Programmierung von BASIC unter Berücksichtigung der besonderen Fähigkeiten von OMIKRON-BASIC. Der Neuling erfährt hier vom ersten Einzeiler bis zu abstrakten Datentypen alle Möglichkeiten der strukturierten Programmierung. Anhand zahlreicher Beispiele mit Flußdiagrammen kann man gute Programmierung nachvollziehen und trainieren.

Im 2. Hauptteil werden Programmier-Kenntnisse erweitert und vertieft. Ausführlich beschriebene und erklärte Programme wie z. B. Fakturierung, ein Kopierprogramm, eine universelle random-access-Datenverwaltung sowie Beispiele zur Betriebssystemprogrammierung vermitteln Neulingen und dem Fortgeschrittenen die Fähigkeit zur perfekten Programmiertechnik.

Im GROSSEN OMIKRON-BASIC-BUCH wird besonderer Wert darauf gelegt, die sinnvolle Anwendung der BASIC-Befehle im Zusammenhang zu erlernen.

Im Anhang findet der Leser die ASCII-Tabelle, einen Index, eine vollständige alphabetische Befehlsübersicht, eine ausführliche Worterklärung und weitere nützliche Tabellen.

Ob mit oder ohne Programmiererfahrung – wer OMIKRON-BASIC beherrschen will, braucht dieses Buch. Sämtliche Programme des Buches gibt es auf Programm-Diskette.

INHALT:

▶ Erklärung der Schleifen- u. Programmstrukturen – Primzahlenberechnung – Zahlenraten ▶ Variablentypen und Arrays – Sieb des Eratosthenes – Adresseingabe ▶ Numerische- u. Stringfunktionen ▶ Unterprogramme u. Prozeduren – Rekursionen – Labyrinthsuche ▶ Multitasking in OMIKRON-BASIC ▶ Abstrakte Datentypen – Druckerspooler – Verkettete Listen ▶ Sequentielle Dateiverwaltung ▶ Random-Access-Dateien – Verkettete Listen auf der Diskette ▶ Grafikprogrammierung – Turtlegrafik – Arbeiten mit mehreren Bildschirmen ▶ Betriebssystemprogrammierung – Aufrufen von TOS-Befehlen – Verwenden der GEMLIB – Erweiterte GEMLIB zur Fensterverwaltung ▶ Beispielprogramme – Fakturierung – Copy/Backup-Programm – universelle Adress-Verwaltung etc. ▶ Anhänge – Worterklärungen – alphabetische Befehlsübersicht – Index – Sonstige Anhänge.

Hardcover Über 400 Seiten einschl. Programm-Diskette **DM 59,-**

Preise sind unverbindlich empfohlene Verkaufspreise

Erhältlich bei Ihrem Computer-Händler

Bücher, die man versteht **Heim Verlag**

EDITORIAL

Wie doch die Zeit vergeht. Dies ist doch tatsächlich schon unsere erste Jubiläumsausgabe !!

ST VISION existiert nun bereits ein ganzes Jahr. Vergleicht man diese Ausgabe mit unserem Erstlingswerk aus dem Sommer 1987, so ist doch eine merkliche Leistungssteigerung zu erkennen - und damit meine ich nicht nur den größeren Umfang des Heftes (von anfänglich 20 auf jetzt 80 Seiten).

Wie ihr sicherlich schon gemerkt habt, mußten wir beginnend mit dieser Ausgabe den Preis auf DM 4,50 erhöhen. Steigende Druck- und Materialkosten sind hierfür die Gründe. Diese Preiserhöhung betrifft aber nur die Einzelhefte, den ABO-Preis von DM 30,- haben wir beibehalten.

Unser fester Leserstamm wird in dieser Ausgabe die angekündigten Fortsetzungen unserer beiden Serien 'Tanglewood - Tips & Tricks' und 'Cyber Lehrgang' vermissen. Da uns kurz vor Redaktionsschluß noch einige neue Produkte und interessante Informationen erreichten, entschlossen wir uns kurzerhand aus Aktualitätsgründen die Fortsetzungen auf die nächste Ausgabe zu verschieben. Keine Angst, aufgeschoben heißt nicht aufgehoben !!!

Nun ist der Sommer 1988 auch schon fast wieder vorbei. Die Urlaubszeit neigt sich dem Ende entgegen. Ich weilte diesen Sommer in meinen Urlaub für vier Wochen in New Jersey und konnte mich dort mit eigenen Augen von der ST Misere in den USA überzeugen. Es war schwierig überhaupt Läden zu finden, die Atari Produkte führten. Sowohl in Computerläden als auch in Buchhandlungen, die in den USA neben Computerliteratur meist auch einiges an Software zu bieten haben, waren meist nur Produkte für Apple-, Macintosh-, IBM-, Commodore-, und Amiga-Computer zu finden. Von Atari keine Spur. Doch zum Glück gibt es auch Ausnahmen. Speziell in New Jersey gibt es in Cedar Knolls einen hervorragenden Computerladen, der das Herz eines jeden Atari-Computer-Besitzers höher schlagen läßt. Ein Atari-only-Geschäft vollgepackt mit Atari 8-bit und ST Soft-, bzw. Hardware. Selbstverständlich konnte ich nicht widerstehen und verschwendete einen Teil meines Urlaubsgeldes für ein paar interessante brandneue Programme, die wir in dieser und der nächsten Ausgabe von ST VISION vorstellen werden.

Schade, daß es nicht mehr Läden dieser Art gibt. Da sieht es an der Westküste der USA bestimmt besser aus, insbesondere seitdem Atari die dort ansässige Federated-Ladenkette geschluckt hat.

So, nun wünsche ich Euch noch viel Spaß beim Lesen und Computern. Auf ein erfolgreiches zweites ST VISION-Jahr.

Euer Michael Schütz







IN DIESER AUSGABE

Le Redacteur Seite 12

Auch in Frankreich erfreut sich der ST wachsender Beliebtheit. Wir freuen uns, ihnen heute ein französisches Textverarbeitungsprogramm näher vorstellen zu können.
Le Redacteur stellt so manch anderes Programm dieser Art in den Schatten. Es besitzt eine Reihe ungewöhnlicher und gleichzeitig nützlicher Features, wie z.B. bisher nicht in Textverarbeitungsprogrammen zu findende Statistiken oder nur selten vorkommende Makro-Befehle. Sollte dieses Programm demnächst auch in einer deutschen Version auf den Markt kommen, so wird sich 1st Word Plus wohl warm anziehen müssen.

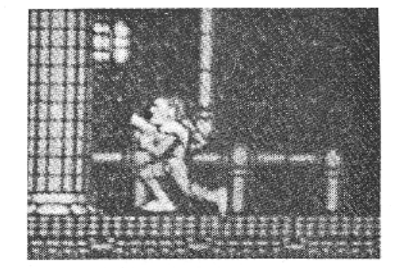
GFA/Omikron-Basic 3.0 Seite 32,46

Der Sommer 1988 entwickelt sich zum Kampf der ST Basic Giganten:

Nachdem es schon seit über einem halben Jahr angekündigt wurde, ist es seit einiger Zeit jetzt erhältlich - GFA Basic 3.0. Über diese neue Version ist schon vor seinem Erscheinen eine Menge geschrieben worden. Kann es halten, was die vielen Vorschauen versprochen haben ?
Wir haben uns das neue GFA Basic vorgenommen und berichten über seine neuen Befehle und Eigenarten.

Auch Omikron präsentierte diesen Sommer die Version 3.0 seines Basic Dialekts. Dieses Basic ist gerade auch deshalb interessant, da es ab sofort in Deutschland beim Kauf eines STs als neues ST Basic mit beiliegt.
Lesen Sie ab Seite 46 was alles verbessert wurde und wie gut der neue Editor wirklich ist.

Der letzte Obliterator Seite 23



Das englische Softwarehaus Psygnosis hat mal wieder zugeschlagen. Ihr neuster Hit Obliterator tritt voll in die Fußstapfen von Hegor dem 'Barbarian'. Wieder einmal ist Grafik und Action total angesagt. Ein Muß für alle Actionfans.

Atari-Allerlei Seite 56

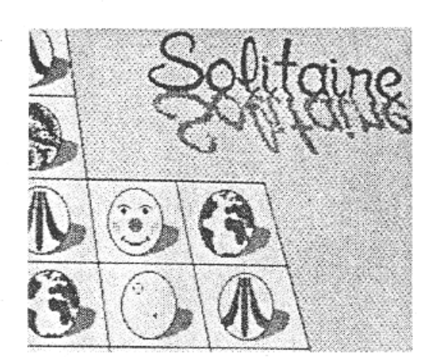
In den USA ging es im Frühjahr und Sommer 1988 rund. Da wurde die komplette Atari Corp. neustrukturiert. Neue Topmanager wurden engagiert und kurze Zeit später wieder entlassen. Wann kann Atari auch in den USA einen Erfolgszug wie in Europa antreten ?

Lesen Sie, was sich alles in den letzten Monaten auf der anderen Seite des großen Teiches getan hat und was für neue Gerüchte derzeit die Runde machen.

Denn die Gerüchteküche ist mal wieder ganz schön am Brodeln. Oberguru Jack Tramiel und die Mannen um Shiraz Shivjy scheinen sich wieder einiges vorgenommen zu haben, um den Computermarkt nicht langweilig werden zu lassen.

PD-Tophits Seite 64

Unsere PD-Liste ist in letzter Zeit auch wieder um einige Disketten gewachsen. Eine Reihe interessanter Programme sind neu hinzugekommen. Ein paar dieser Programme stellen wir ihnen heute auf den Seiten 64 und folgende vor.



Ob Sie einen Fußballverein managen möchten, Vokabeln büffeln müssen, Texte oder Briefe schreiben wollen oder einfach nur spielen möchten. In unserer PD-Library ist für jeden Geschmack etwas vorhanden. Lesen Sie nähere Einzelheiten über unsere neuen Schlager Fußballmanager, Hypervoc, Fuffy, Solitaire (siehe Bild), Bolo, Star Trash, Zarge und Easy Writer in unserem ausführlichen PD-Teil.

Bit, Byte und Bus Seite 58

Da leistungsfähige Computer immer billiger werden, kaufen sich immer mehr Leute Geräte für zu Hause. Oftmals noch absolute Computerneulinge stehen sie jetzt vor dem vom Verkäufer als optimalen und leicht zu verstehneden Einsteigermodell gepriesenen Wundergerät und wissen nicht weiter. ST VISION wird beginnend in dieser Ausgabe einige Grundbegriffe des Computerns erläutern.

SOFTWARETESTS

PROGRAMMIERSPRACHEN

RELAX

STÄNDIGE RUBRIKEN

INTERNATIONALE NEWS

Rund um den Atari

IFA Köln: Datenbank für Börsenkurse

Die Firma IFA Köln, bekannt durch ihr Börsenspekulationsprogramm 'JAMES', bietet für alle Kunden nun eine spezielle Datenbank an. Hierdurch haben die Besitzer von 'JAMES' die Möglichkeit, sich automatisch die aktuellen Börsenkurse per Telefon und Modem übermitteln zu lassen.

Der Zugang erfolgt direkt von 'JAMES' aus, ohne daß sich der Anwender mit Problemen der Datenfernübertragung beschäftigen muß. 'JAMES' erkennt nach erfolgreicher Verbindungsaufnahme die neuen Börsenkurse und verarbeitet sie ebenfalls automatisch - es ist also optimale Anwenderfreundlichkeit gegeben. Der Datenbankzugang erfolgt mit einer Übertragungsgeschwindigkeit bis zu 2400 Baud. Hierbei können rund 120 Aktienkurse pro Minute übermittelt werden.

Die Datenbank steht dem Anwender 24 rund um die Uhr zur Verfügung und wird täglich mit neuen Daten gefüttert. IFA Köln bietet diesen Service für monatlich DM 25.- an.

Geplant ist außerdem eine Wirtschaftsdatenbank mit Hintergrundinformationen über Wirtschaftsunternehmen. Hierbei können Daten wie Inhaber, Beteiligungen, Konkurse, Aufsichtsrat, Produktpalette usw. abgefragt werden. Eine neue Version von 'JAMES' ist ebenfalls in Sicht. Etwa im Herbst wird 'JAMES 2' zu erwarten sein.

IFA Köln
Gutenbergstr. 73
5000 Köln 30
0221/520428

Tommy Software, Berlin

Bei der Firma Tommy Software wird derzeit an einem Programm namens 'Sound Machine ST' gearbeitet. Hierdurch können digitalisierte Sounds weiterverarbeitet werden, wobei die Benutzerführung an 'Musix 32' angelehnt ist. Die Qualität der Samples ist vergleichbar mit dem Sound von Goldrunner. Der Preis wird etwa bei DM 98.- liegen, wobei eine Demoversion zur Atari-Messe in Düsseldorf vorliegen dürfte.

Passend hierzu ist ein A/D-Wandler in Vorbereitung, um eigene Samples aufnehmen zu können.

Tommy Software
Selchowstr. 32
1000 Berlin 44
030/6214063

CORRUPTION

Ein neues Super - Adventure von Rainbird/Magnetic Scrolls

Jedem Adventureliebhaber sind die Spiele 'The Pawn', 'Guild of Thieves' und 'Jinxter' bekannt - Ein Inbegriff für extrem gute Grafiken und eine gute Story.

Der Programmierer von 'The Pawn', Rob Steggles, hat nun ein weiteres Werk geschaffen. Die Handlung spielt diesmal aber nicht in Kerovnia, sondern in unserer Zeit. In 'Corruption', so der Titel des Adventures, dreht es mehr um weltliche und materielle Dinge wie Geld, Macht und Intrigen, wobei Attentate und Autobomben natürlich nicht fehlen.
'Corruption' unterscheidet sich aber auch von der Spielweise erheblich von seinen Vorgängern. Es ist praktisch unmöglich, das Spiel auf Anhieb zu lösen. Vielmehr gilt es die Verhaltensweisen von etwa 30 Personen zu ergründen, daraus die richtigen Schlüsse zu ziehen und entsprechend zu handeln.

Eine neue Spielidee die vielleicht nicht jedem Gefallen wird, aber den Spielwert von 'Corruption' erheblich steigert. Wer aber lieber auf konventionelle Art und Weise ein Adventure lösen möchte, der kann auf den nächsten Hit von Magnetic Scrolls warten: 'Fish' wird in Kürze erscheinen.

NEUIGKEITEN VON OMIKRON.

Neue Produkte und Libraries

Viele Neuigkeiten gibt es aus dem Hause OMIKRON zu berichten. So sind für das Omikron Basic weitere Programmierhilfen dazugekommen. Zu den bisher erschienen Libraries Statistik, Numerik, Midi und Masken-Editor kommen gleich 4 weitere hinzu. Dies sind die Complex-Library, die Finanzmathematik-Library, eine Easy-GEM-Library und eine Assembler-Library.

Complex-Library

Für Berechnungen aller Art mit komplexen Zahlen. Dies reicht von der normalen Additionen bis zur Matrizenrechnung. Die Speicherung der komplexen Zahlen erfolgt in einem String, so daß ein direkter Zugriff problemlos möglich ist. Diese Library ist zu einem Preis von DM 79.- erhältlich.

EasyGEM-Library

Hierdurch wird der Zugriff auf die Gem-Komponenten unter Omikron Basic erheblich vereinfacht. Für DM 79.- wird diese Library angeboten.

Finanzmathematik-Library

Verzinsungen aller Art, Rentenrechnung und Datumsberechnungen zum Einbau in eigene Omikron Basic Programme. Insgesamt stehen über 240 Funktionen zur Verfügung, sogar engl. und franz. Zinsrechnungen sind implementiert. Diese Library wird für DM 129.- angeboten.

Assembler-Library

Ermöglicht die direkte Verwendung von Inline-Assembler-Code mit Mnemonics und Labels innerhalb des Basic-Programmes. Diese Library ist für den Interpreter und Compiler verwendbar.

2nd SLM

Hiermit können 2nd Word Dokumente auf dem Atri SLM 804 Laser ausgegeben werden.

Link-it

Nun arbeitet Omikron Basic auch mit Hochsprachen und Assemblern zusammen. Mit 'Link-it' ist es möglich, Objekt-Code von C-Compilern oder Assemblern mit compilierten Omikron Basic Programmen zu verbinden. Der Verkaufspreis liegt bei DM 79.-.

Das OMIKRON.Buch für Einsteiger

Dieses Buch richtet sich an den absoluten Einsteiger in die Programmiersprache Omikron Basic. Der Anfänger bekommt hierbei Einblick in die grundlegene Arbeitsweise des Computers und des Omikron Basic-Interpreters. Die Thematik endet etwa bei der Programmierung von Schleifenstrukturen in Basic. Hardcover, 200 Seiten, DM 39.-.

OMIKRON.Turbo-Assembler

Schnellster Assembler für den Atari ST. Es werden Übersetzungszeiten von etwa 250.000 Zeilen in der Minute erreicht. Die maximale Geschwindigkeit liegt bei etwa 2,4 Millionen Zeilen pro Minute.

Im Assembler integriert sind ein Editor und ein Debugger. Der Editor ist ähnlich wie Tempus aufgebaut und testet die eingegebenen Zeilen sofort auf ihre Richtigkeit. Gleichzeitig erfolgt eine automatische Formatierung der Zeile und eine Übersetzung der Mnemonics in Token. Der Sourcetext wird bei Speicherung auf Diskette komprimiert, so daß Diskettenplatz gespart wird. Die Tastaturbelegung des Editors ist frei definierbar.

Der eingebaute symbolische Debugger ist eine Weiterentwicklung des im IDEAL integrierten Debuggers. Implementiert wurden auch die Funktionen 'Untrace' und 'Quit If'.

Der Omikron Assembler kann Sourcetexte von anderen Assemblersystemen einladen und automatisch eine Formatanpassung und Befehlsumwandlung vornehmen. Dadurch können Sourcetexte, die mit Profimat, Devpac oder IDEAL geschrieben wurden, direkt verwendet werden.

Der Omikron Assembler wird in zwei Versionen angeboten. Für Anfänger steht eine Einsteigerversion zur Verfügung. Diese ist zu einem Preis von nur DM 99.- erhältlich. Für Fortgeschrittene wird eine makrofähige Version angeboten. Diese kann lokale Labels verarbeiten, beherrscht die bedingte Assemblierung, Include-Dateien und Assemblerschleifen. Die Makroversion wird zu einem Preis von DM 198.- angeboten.

A-MAGIC TURBODIZER

Die Version 2.0 ist nun erhältlich

Der A-Magic TurboDizer gehört sicher zu den Spitzenmodellen auf dem Markt. Leider war die Software bisher noch nicht ganz ausgereift, die Version 2.0 bietet dagegen viel Neues.

Zum Speichern der Bilder stehen nun alle gängigen Formate zur Verfügung: Degas, Art-Director, Neochrom, Fleet Street Publisher, Colorstar, Monostar, Stad, Doodle, Bitmap. An der Benutzeroberfläche wurde ebenfalls einiges verändert. Im Diskmenü stehen umfangreiche Diskettenoperationen zur Verfügung. Es können Ordner angelegt und gelöscht werden oder der freie Speicherplatz der Diskette ausgegeben werden. Um den Diskettenplatz optimal zu nutzen, kann eine Diskette bis ca. 850Kb formatiert werden. Um mehrere digitalisierte Bilder in einer Sequenz abzuspeichern, wurde die Funktion Animate implementiert.

Mit ihr können kleine Trickfilmsequenzen erstellt und abgespeichert werden. Ein besonderer Gag ist das im Lieferumfang enthaltene Accessoire 'TV.ACC'. Hiermit es möglich, im Desktop die Sportschau zu verfolgen, ohne die Arbeit unterbrechen zu müssen. In einem auf dem Desktop beliebig plazierbaren Fenster werden Fernsehbilder nahzu in Echtzeit digitalisiert, wodurch sich ein bewegtes Bild ergibt. So etwas hat es sicher auf noch keinem anderen Computersystem gegeben!

ATARI:

Eine gute und eine schlechte Nachricht

Preiserhöhung

Nichts ist auf dem Computermarkt zur Zeit so wertvoll wie dynamische Speicherbausteine. Japanische Produzenten können und wollen ihre Kapazitäten nicht ausweiten, so daß in den letzten Monaten eine zunehmende Verknappung der Bausteine zu verzeichnen ist. Die Folge davon sind steigende Preise, die sich natürlich auch auf die Computer auswirken. Atari kündigt aus diesem Grunde eine Preiserhöhung noch für dieses Jahr an.

SPF004

Kein Geheimcode für das amerikanische Pentagon, sondern die Typenbezeichnung für eine Koprozessorkarte für die Mega ST's. Bestückt ist die Karte mit dem 68881 Floating-Point-Processor von Motorola, der mit einer Taktfrequenz von 16 MHz betrieben wird. Vewendet wird der 68881, um mathematische Berechnungen zu beschleunigen. Voraussetzung hierfür ist, daß die Software den Zusatzprozessor unterstützt. Der Preis wird bei etwa DM 400.- liegen.

TEMPUS VERSION 2.00

Nach vielen Hinauszögerungen ist nun endlich seit Mitte August ein neues Update von Tempus erhältlich. Es handelt sich hierbei um den bekannten Texteditor, der in erster Linie zur Erstellung von Quellprogrammen (Listings) gedacht war. Nun gibt neben den schon bekannten Fähigkeiten die Möglichkeit, Fließtexte zu verarbeiten.

Weiterhin können Zeilen sortiert, Rechenergebnisse des eingebauten UPN-Rechners in den Text übernommen werden, Sie können die Speicheraufteilung (Mindestsystemspeicher und Maximalarbeitsspeicher von Tempus) eigenständig vornehmen, sämtliche Installationen sind Gruppenweise getrennt (Grundparameter, Druckertreiber, Tastatur- und Funktionstastenbelegung).

An die Möglichkeit eines unbedachten Absturzes Ihres Rechners wurde auch gedacht. So kann ein automatisches Abspeichern der Texte, die von Ihnen gerade bearbeitet werden, in einstellbaren Zeitintervallen erfolgen. Auch eine einstellbare Bildabschaltung (Bildschirmschwarzstellung) ist vorhanden.

Die Verarbeitungsgeschwindigkeit ist weiterhin unvergleichlich: Die durchschnittlichen Bildschirmaufbauten bei maximaler Fenstergröße werden mit 23 pro Sekunde angegeben. Deshalb gibt es sogar einen einstellbaren Verzögerungswert beim Rollen und Blättern von Fensterinhalten.

Tempus 2.00 ist bis zum 31 Oktober zum Einführungspreis von DM 109.-, danach von DM 129.- bei CCD in Eltville zu beziehen.

CCD - D.Beyelstein
Postfach 175
6228 Eltville
Tel. 06123-1638

fibuSTAT

Tabellenkalkulation auf Berechnungsgrundlage der Gewinn- und Verlustrechnung (GuV)

fibuSTAT

Rechtzeitig zur Atari-Messe in Düsseldorf wird das Programm fibuSTAT von der Firma Prodata in einer endgültigen Version zur Verfügung stehen.

FibuSTAT ist zur Unterstützung der Überprüfung und Überwachung der Unternehmensentwicklung erstellt worden. Das Programm ermöglicht es, in übersichtlicher Form (grafische und tabellarische Übersicht) einen Einblick in die Geschäftsentwicklung der Unternehmung zu erhalten. Dabei wurde ein besonderes Augenmerk auf eine benutzerfreundliche Anwendung gelegt, die in Zusammenarbeit mit einer hohen Arbeitsgeschwindigkeit eine effiziente Verwaltung der zur Verfügung stehenden Informationen erlaubt.

Die eingegebenen Daten können mit Hilfe von Balken-, Linien- und Kuchen-Diagrammen veranschaulicht werden (siehe Bild). Die Eingabe neuer Daten erfolgt direkt innerhalb der Tabelle, die dem Benutzer Auskunft über die Entwicklung und den momentanen Stand des Betriebs gibt. Weitere Informationen sind durch Kennzahlen wie Cash-Flow und Gesamtkapitalrentabilität erhältlich. Zur besseren Übersicht kann die Tabellenausgabe in verschiedenen Schriftgrößen erfolgen.

Die Ausgabe der Tabelle und der Grafiken kann wahlweise auf dem Bildschirm oder dem Drucker erfolgen. Im Lieferumfang sind Druckertreiber für Epson FX (9 Nadeln) und Nec P6 (24 Nadeln) enthalten.

Interessant ist die Nutzung von fibuSTAT in Zusammenhang mit fibuMAN, zu dem eine Schnittstelle besteht. Hier können die monatlichen Abrechnungen von fibuMAN in fibuSTAT eingelesen werden.

Im Lieferumfang von fibuSTAT ist eine einseitig bespielte Diskette und ein ausführliches deutsches Handbuch enthalten. Die Firma Prodata bietet fibuSTAT für DM 398.- an.

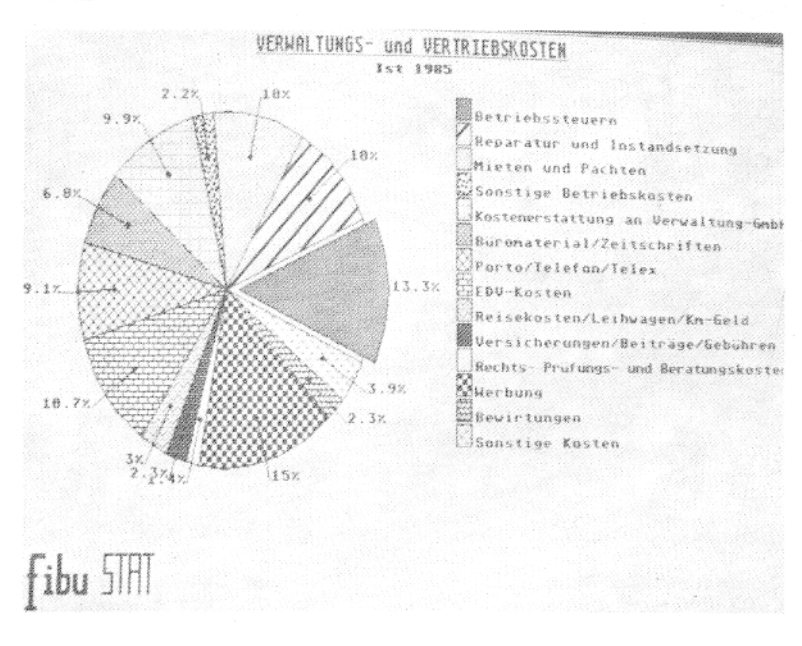
fibuMAN

Das Finanzbuchhaltungsprogramm fibuMAN wird zu der Atari-Messe in der neuen Version 3.0 erhältlich sein. Der Preis der neuen Version bleibt aber unverändert. Immerhin sind von diesem System schon ca. 1000 Implementationen in einem dreiviertel Jahr zu verzeichnen. Geplant sind außerdem Versionen für Amiga, Apple und Unix. Eine MS-Dos Version ist bereits lieferbar.

Folgende Änderungen ergaben sich in der Version 3.0:

- Einzelbuchungen sind bis zu DM 9.999.999,99 möglich
- Vollautomatische Datenregeneration nach Stromausfall o.ä.
- Anzeige der Kontensalden, die sich nach einer Buchung ergeben würden, bereits vor der Buchung (was wäre wenn)
- Optionales akustisches Signal bei und unmittelbar vor "negativem Kassenbestand"
- Erweiterte Druckersteuerung
- Skontierungsautomatik
- Probedruck zwecks Papierjustierung vor Ausdruck der Umsatzsteuervoranmeldung auf das Formular des Finanzamts
- Vollautomatische Wareneinsatzbuchung bei Bilanzierung
- Optionales automatisches Hochzählen der Belegnummer
- Zusätzliche Umsatzsteuersätze

Prodata
Broicherstr. 39
5060 Bergisch Gladbach 1
Tel. 02204/51456, 0161/221579



Glasnost und Atari

Der Herausgeber des amerikanischen Computermagazins PC World wird als erster ein Personal Computer Magazin in der UdSSR vertreiben - in russischer Sprache !!!

Die erste Ausgabe ist bereits im August mit einer Auflage von 50.000 Heften erschienen. Der Preis für ein Heft soll bei umgerechnet ca. DM 6,- liegen.

Die Zeitschrift soll zu einem Drittel aus Informationen über die Entwicklung der Computerszene in der Sowjetunion bestehen. Die restlichen Seiten sollen mit Berichten über den amerikanischen und den westeuropäischen Markt gefüllt werden.

Zur Zeit ist die Zahl der PC Benutzer in der Sowjetunion noch relativ klein und nur wenige Informationen über computermäßige Neuentwicklungen werden an die breite Öffentlichkeit weitergegeben.

Ca. 20 % des 150 Seiten starken Magazins sind Anzeigen. Unter den Firmen die in der ersten Ausgabe inserieren finden sich so prominente Namen wie Ashton Tate, Siemens, Microsoft und ... ATARI !!!

Ein hier nicht genannt möchtender Atari Offizieller ließ auf die Frage, was sich Atari von dieser Werbung verspreche, folgendes verlauten:

Eines der vielen Langzeitziele von Atari sei es, daß bald in jedem russischen Klassenzimmer mindestens ein ST steht und das Bild von Jack Tramiel gleich neben denen von Lenin und Genossen Gorbatschow an der Wand hängt. mts

Quelle für diesen Artikel:
Current Notes Vol.8 - No. 6

Slaygon - neues Grafik-/Actionadventure aus dem Hause Microdeal

Wer erinnert sich nicht an die ausführlichen Berichte in unseren letzten beiden Ausgaben über das Microdeal Adventure Tanglewood. Nun mittlerweile gitb es bereits ein neues Programm aus besagtem Hause.

Es heißt Slaygon und sieht Tanglewood auf den ersten Blick sehr ähnlich. Die Aufmachung ist im gleichen Stil wie die seines Vorgängers gehalten - eine bunte Spiele Box, zwei Disketten, eine kleine schwarz-weiße Spielbroschüre und die obligatorische 'Registration Card'. Betrachtet man sich das ganze etwas näher, so kann man Slaygon als eine Mischung zwischen Tanglewood und dem etwas älteren Spiel Hacker betrachten.

Ziel des Spiels ist es mit Hilfe des Slaygons, dem neusten und besten miltiärischen Roboter, das Cybrodynamics Labor zu zerstören, wo ein tödlicher Virus entwickelt wird. Sie übernehmen natürlich die Fernsteuerung des Roboters und müssen sich nun auf den Weg machen, das Labor, das aus ca. 500 Räumen und 5 Meilen an Gängen besteht, zu erforschen.

Das Spiel ist komplett in GFA Basic gechrieben und wird in den USA für ca. US$ 40 verkauft. ST VISION wird Slaygon in der nächsten Ausgabe in einem ausführlichen Testbericht näher vorstellen.

mts

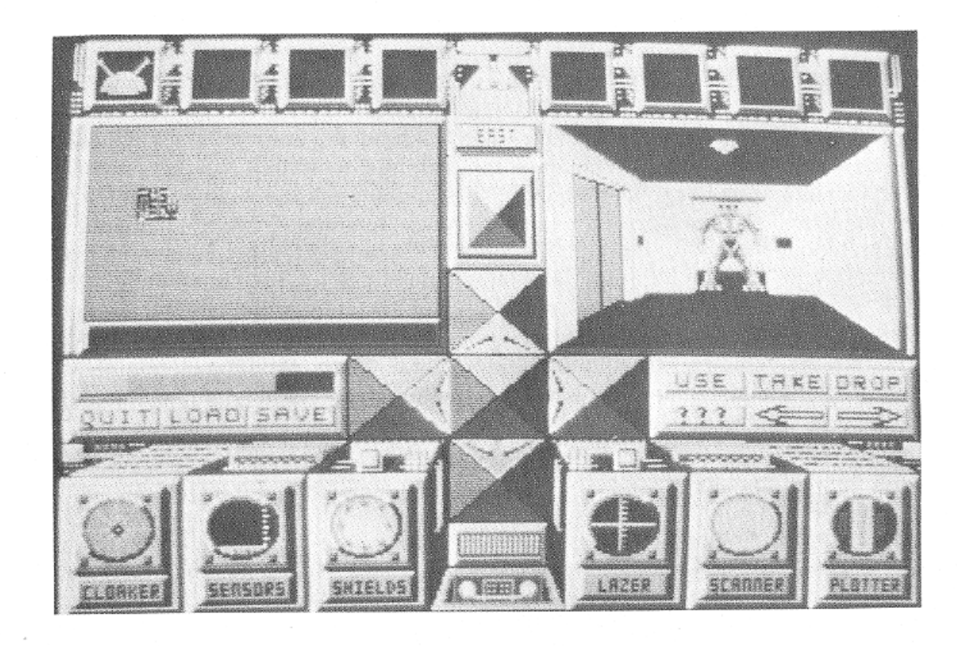


BILD: Slaygon im Labyrinth des Forschungslabors

Atari in Europa - eine kleine Erfolgsstatistik

In der Bundesrepublik machen Atari Computer 10 % des gesamten Marktes für Computer über DM 1.000,- aus.

In der Schweiz beträgt der Anteil der STs im Erziehungsbereich 30 %.

In Großbrittanien machen die Atari Computer Verkäufe zur Zeit 40 % des 16-bit Computermarktes aus.

Von allen ST Computern wird der 1040 ST am häufigsten für MIDI-Anwendungen eingesetzt.

3D Break-Thru dreidimensionales Breakout

Wer kennt nicht Breakout, einen der Klassiker unter den Computerspielen. Für den ST gibt es ja mittlerweile eine Vielzahl an Breakout-Varianten.

Die beiden bekanntesten dürften wohl Arkanoid und Bolo! sein. Der ANTIC/START Softwarevertrieb 'THE CATALOG' hat seit kurzem eine weitere interesante Variante dieses Urspiels im Angebot. Es heißt 3D Break-Thru und ist, wie der Name es schon verrät, eine Art dreidimensionales Breakout.

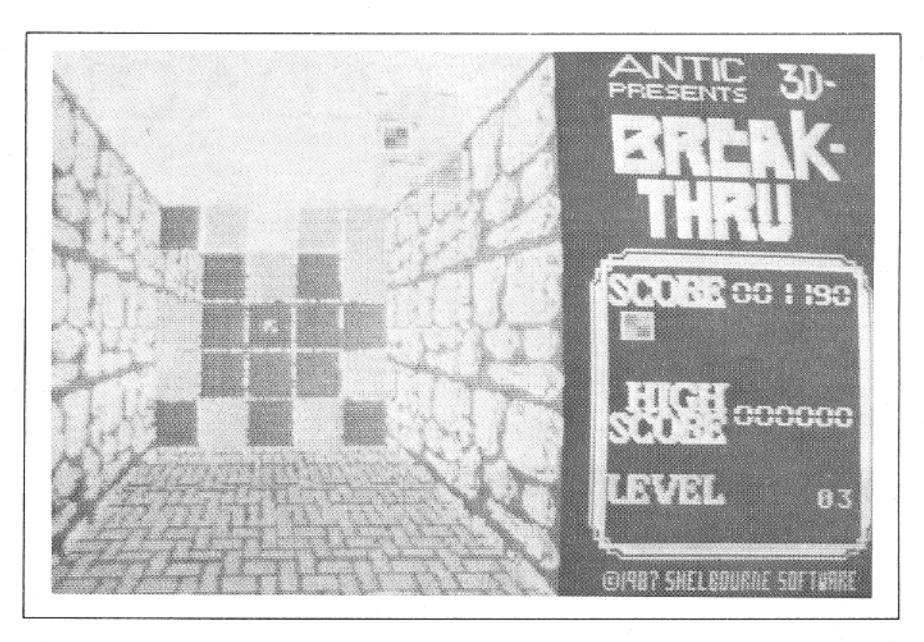


BILD: 3D Break-Thru - Unordnung in der Bibliothek

Der Spieler befindet sich in einem 3D Labyrinth und muß eine Barriere nach der anderen mit Hilfe seines 'Paddles' durchbrechen. Je weiter er in das Labyrinth vordringt, um so schwieriger wird das Vorwärtskommen. In den etwas komplexeren Gängen werden die herumschwebenden Gegenstände immer unfreundlicher. Man besitzt dann zwar die Möglichkeit diese abzuschießen. Allerdings wird mit jedem Schuß das 'Paddle' kleiner.

Das Spiel funktioniert auch mit der bisher in Deutschland relativ unbekannten Stereotek 3D Brille für den ST. Aber auch ohne diesen Hardwarezusatz ist der 3D Effekt schon relativ gut. In den USA kostet das Programm US$ 30-40. Bisher ist der ST VISION Redaktion kein Vertrieb in Deutschland bekannt, der dieses Programm in sein Verkaufsreportoir aufgenommen hat. Zum Betrieb benötigt man mindestens einen 520 ST mit Farbmonitor.

mts

Carrier Command - neues aus dem Hause Rainbird

Dieses neue Produkt für den ST ist schon seit Monaten durch Werbung angekündigt worden. Jetzt ist es endlich erhältlich.

Carrier Command ist ein Strategiespiel, das auch Arcade-Game-Elemente beinhaltet. Ihre Aufgabe ist es, eine an Bodenschätzen reiche Inselgruppe mit Hilfe ihres modernen 'Flugzeugträgers' zu besetzen und nutzbar zu machen, bevor sie einer fremden Macht in die Hände fällt.

Ausführlicher Testbericht folgt in der nächsten Ausgabe. mts

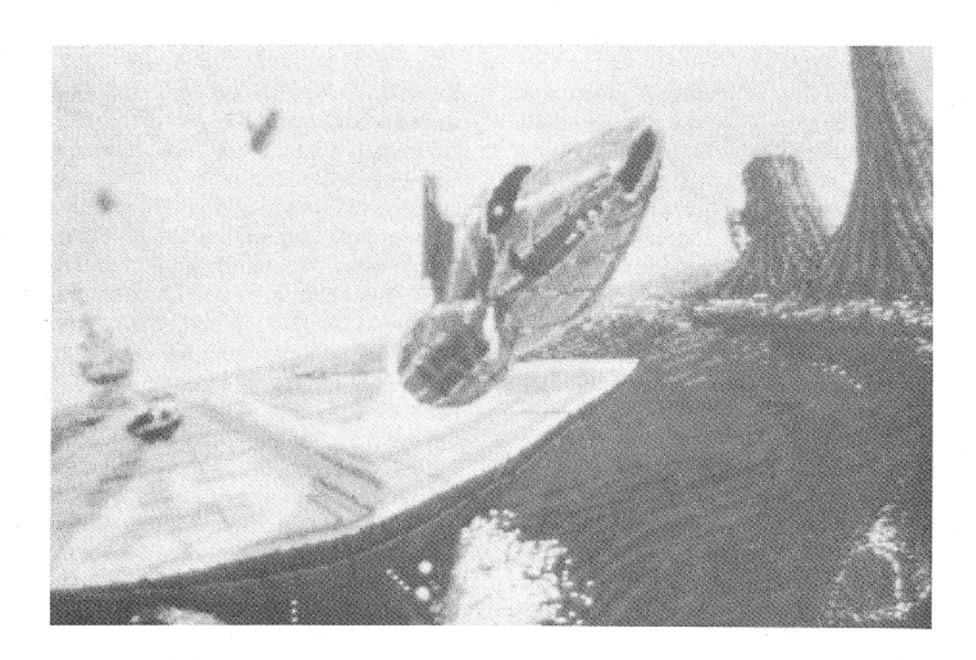


BILD: Titelbild von Carrier Command

LE REDACTEUR

Textverarbeitung auf französich

Aus Frankreich kommt ein neues beachtliches Textverarbeitungsprogramm für die ST Computer, Le Redacteur. Geschrieben wurde es von Dominique Laurent, Jean-Christophe Lejeune und Jean-Michael Buraux. Laurent ist ebenso der Autor von Turbo-Dos, einer Verbesserung der TOS-Routinen. Man kann sagen, daß das 'Le Redacteur' von der Qualität her mit Tubro Dos gleich zu setzen ist.

Auf den ersten Blick sieht das Programm aus wie 1st Word Plus, aber es benutzt keine GEM-Routinen. Es ist das schnellste Textverarbeitungsprogramm, das ich bisher auf dem ST gesehen habe. Obwohl es ein eigenes neues Textformat (.LIB) verwendet, ist es dennoch zusätzlich in der Lage Texte im GST Format (1st Word Plus) zu laden und auch abzuspeichern.

Der Anfangsbildschirm von 'Le Redacteur' hat gewisse Ähnlichkeit mit 1st Word Plus. So findet man auch am unteren Bildschirmrand eine Leiste bestehend aus den rautenförmigen ST-Funktionstasten, über die z.B. die Schriftaufbereitung (Bold, Underline, Italic, Light) eingestellt werden kann. In der Mitte des Bildschirms sieht man den zur Verfügung stehenden Zeichensatz - 1st Word Plus läßt grüßen.

Über dem geöffneten File sieht man selbstverständlich auch wieder das Lineal mit den TABs. Mittels eines Doppelklicks läßt sich dieses in eine Statuszeile verwandeln, in der dann die aktuelle Zeilen- und Spaltennummer, die Anzahl der Zeichen, das Datum und die Uhrzeit eingeblendet werden. Da für das Desktop ein relativ kleiner Zeichensatz verwandt wurde, empfindet man die blinkende Sekundenanzeige meines Erachtens nicht als störend.

Alle Befehle können über Menüs mit Hilfe der Maus abgerufen werden. Es stehen einem eine ganze Reihe von Befehlen zur Verfügung. Wir wollen unseren Streifzug durch die Menüs in der oberen linken Ecke beginnen:

Hinter INFORMATION verbirgt sich diesmal nicht nur der Hinweis auf die Autoren des Programms. Man kann hier sowohl den verbleibenden Speicherplatz des Systems und der Diskette ersehen, als auch Datum und Uhrzeit einstellen.

Zu FILE OPERATION ist zu sagen, daß man Texte in ASCII, GST und dem Programm eigenen LIB-Format laden und abspeichern kann. Die Geschwindigkeit mit der selbst lange Files eingelesen werden, ist dabei sehr beachtlich - es kommt zu keinen von 1st Word Plus her gewohnten Verzögerungen.

Ein komplettes weiteres Menü verbirgt sich unter DISK OPERATIONS. Hier können Files gelöscht, umbenannt und kopiert werden. Die Kopie erhält dabei einen eigenen Namen. Man kann auch neue Ordner erzeugen, in die man dann Files kopieren kann, wobei das alte File gelöscht wird.
Selbstverständlich kann man auch Disketten formatieren. Das Programm fügt den Filenamen die jeweiligen Extender automatisch hinzu. Diese können in einem eigenen Menü vordefiniert werden.

Das nächste Menü heißt EDITING. Für die Such-bzw. Ersetzfunktion steht ein sage und schreibe 57-Zeichen langer String zur Verfügung. Hierbei kann auch nach Schriftaufbereitung gesucht werden. Die Suchrichtung kann aufwärts oder abwärts ausgehend von der Cursorposition gewählt werden - oder man läßt einfach den kompletten Text durchsuchen.
Auch hier fällt wieder die Geschwindigkeit auf: Das Ersetzen aller 'a's durch 'e's in einem 10-seitigen Dokument dauert nur wenige Sekunden.

'Le Redacteur' besitzt zwar nicht die Lichtgeschwindigkeit von Tempus, er ist aber dennoch sehr schnell. Textblöcke können in sekundenschnelle definiert und verschoben werden. Ganze Texte können blitzschnell neu formatiert werden. Selbst ein 100 Seiten langes File ist in wenigen Sekunden von der Harddisk eingeladen. Bei 1st Word Plus dauert der gleiche Vorgang geschlagene 80 Sekunden. Gar nicht auszudenken wie lange da das Laden von einer normalen Floppy dauern würde.

Auch beim Neuformatieren von Text wird die schnelle Verarbeitung deutlich. Ein Mausklick auf die Zentrierfunktion und sofort ist der komplette Text zentriert. Das gleiche gilt für rechts- und linksbündiges Neuformatieren.

Textmarkierungen können gesetzt und wieder entfernt werden. Alle Schriftaufbereitungsformen von 1st Word Plus sind vorhanden und können über die Funktionstasten ein- bzw. ausgeschaltet werden (auch Superscript und Subscript).

Zusätzlich verfügt 'Le Redacteur' über zwölf verschiedene Zeichensätze. Ein Absatz darf allerdings aus nicht mehr wie einem Zeichensatz bestehen.

Zur Textblockverarbeitung ist zu sagen, daß ein Block angefangen von einem Buchstaben bis zum kompletten Text groß sein kann. Äußerst praktisch ist auch die Möglichkeit, Textblöcke ohne vorheriges Abspeichern ausdrucken zu können.

Die Druckertreiber werden über ein Menü ausgewählt. Ein Hex-File kann vom Programm aus konfiguriert werden. Folgende Druckertreiber werden standardmäßig mitausgeliefert: Brother 1409, Epson LX, KX P1080, Laserjet Atari SMM80 und SLM804.

Die Hex-Files sehen den 1st Word Plus Files sehr ähnlich und können als ASCII Texte geladen und verändert werden. Nennt man seine eigene Druckeranpassung IMPRIME.CFG (PRINT.CFG), so wird beim nächsten Booten genau wie bei 1st Word Plus dieses File installiert.

Das nächste ist das OPTIONS Menü. Hier findet man so feine Sachen wie z.b. Macros. Ein absolutes Manko fast aller Texverarbeitungsprogramme für den ST war das Fehlen dieser Funktion. Bis zu 36 Tasten können mit bis zu jeweils 78 Zeichen langen Strings belegt werden. Dies verschafft einem einen Macropuffer von 2.800 Zeichen. Das ist wirklich beeindruckend.

Eine weitere Option besteht darin, die Tastaturbelegung zu verändern. Hierzu stehen einem auch eine Reihe von bisher nicht bekannten Zeichen zur Verfügung (Linien, Ecken, etc.).

Im PARAMETRER Menü können einige interessante Einstellungen vorgenommen werden, die auch als neue Standardwerte abgespeichert werden können. Hierzu gehört das Einstellen einer Art File-Schutzfunktion. Man kann eine Zeitspanne von einer Sekunde bis zu 99 min und 99 Sekunden eingeben. Drückt man jetzt länger als vorher eingestellt keine Taste, so wird automatisch, der zur Zeit im Speicher befindliche Text abgespeichert. Natürlich unter Verwendung eines eigenen Extenders, so daß das Originalfile nicht überschrieben wird.

Die letzte Funktion ist gleichzeitig die erstaunlichste. Hier kann man nämlich einen Textblock oder den gesamten Text analysieren lassen. Ein zehn Seiten langer Text war sofort analysiert. Als erstes erhält man die Anzahl der Buchstaben, Worte, Zeilen, Sätze und Absätze des Textes. Desweiteren zeigt eine Grafik die durchschnittliche Wortlänge (in Buchstaben) und Satzlänge (in Worten) an. Wenn sie wissen wollen, wie oft irgendein Wort in ihrem Text vorkommt, null Problemo !!! KLckien Sie einfach 'OCCURANCE DE MOTS' an. Sofort erscheint eine kleine Sanduhr auf dem Bildschirm. Sobald der Sand vollständig im unteren Teil angekommen ist, wurde ihr kompletter Text durchforstet. Bei einem zehn seitigen Text dauerte dies 2 Sekunden, bei 90 Seiten fast 300 Sekunden. Sie erhalten dann einen Index aller verwandten Worte, der entweder alphabetisch oder nach Häufigkeit sortiert ist. Diese Indexliste kann auch als File abgespeichert werden.

Das Schreiben mit 'Le Redacteur' ist sehr angenehm. Beim Zeilenumbruch kommt es zu keinen Verzögerungen. Bildschrimscrolling und Cursorsteuerung sind sehr schnell. Man kann sowohl vorwärts wie rückwärts um einen Buchstaben, ein Wort, einen Satz, einen Paragraph oder eine Seite springen.

Das Programm ist nicht kopiergeschützt, besitzt aber eine Art Käuferschutz. Gelegentlich taucht eine Nachrichtenbox auf, die nach einem bestimmten Wort an einer bestimmten Stelle im Handbuch fragt. Kann man diese Frage nicht richtig beantworten,. so bleibt einem nichts weiter übrig als einen Reset durchzuführen. Seinen Text kann man dann vergessen. Die Abfrage erscheint bei häufigem Benutzen vielleicht alle zwei oder drei Tage einmal.

Was kann 1st Word Plus, was 'Le Redacteur' nicht kann ? Schreibung mit Bindestrich (Hyphenation), Grafikeinbindung, Fußnotenverwaltung und Rechtschreibungsüberprüfung. Inwieweit diese fehlenden Merkmale für den Einzelnen wichtig sind, muß jeder für sich entscheiden.

Das Programm existiert bisher nur in seiner französischen Version. Es ist erst seit ein paar Wochen auf dem Markt. Wenn Sie mit 1st Word Plus nicht zufrieden sind, ein bischen französich kennen und ein einfaches Französischlexikon besitzen, dann sollten Sie sich dieses exellente Textverarbeitungsprogramm sofort kaufen.

Mir sind bisher nur zwei Bezugsquellen für 'Le Redacteur' bekannt:

1. Coconut Republique
   13 Boulevard Voltaire
   75011 Paris
   Telef.: 43 55 63 00

2. Ultima
   5 Boulevard Voltaire
   75011 Paris
   Telef.: 43 38 96 31

Preis: ca FF 600 (=DMxxx)

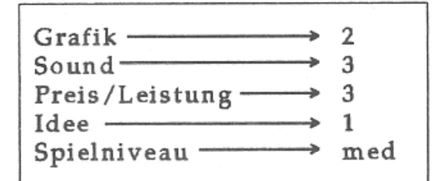
Andreas Ramos, Aarhus (DK)

MANIAX

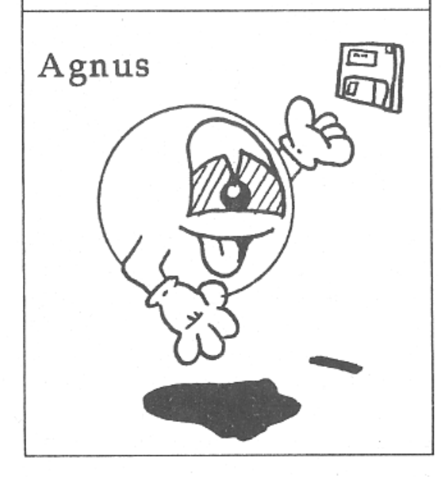
Spannung, digitalisierter Sound und tolle Grafiken made in Germany

Maniax, Programmiert von Marc Ebner, Kingsoft GmbH, DM 49.95

Wer bei Maniax erfolgreich abschneiden möchte, sollte den Beruf des Joystickakrobaten gelernt haben. Hier sind schnelle Reaktionen, gute Nerven und eine gehörige Portion Mut



Grafik	2
Sound	3
Preis/Leistung	3
Idee	1
Spielniveau	med



gefragt, um nicht ein Opfer des Drachens oder der Totenköpfe zu werden.

Die dem Spiel Maniax zugrundeliegende Spielidee stammt vom Arcade-Klassiker Qix - damals ein großer Erfolg in den Spielhallen. Erstmals wurde von dem Prinzip abgelassen, in kürzester Zeit möglichst viele Weltraummonster zu vernichten.

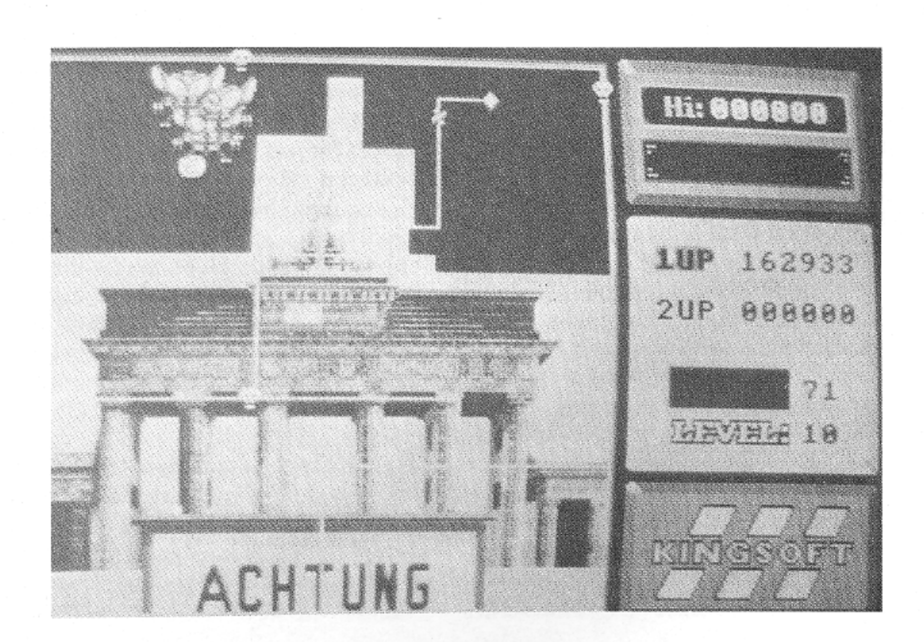
Ziel ist es, aus dem rechteckigen Spielfeld eine Fläche von mindestens 75% auszuschneiden. Dazu steht dem Spieler ein Sprite zur Verfügung, das nur in horizontaler und vertikaler Richtung bewegt werden kann, so daß auch nur eckige Flächen ausgeschnitten werden können. Außerdem kann sich der Spieler nur jeweils am Rand seines Spielfeldes bewegen, das sich durch jedes neu ausgeschnittene Feld verkleinert.

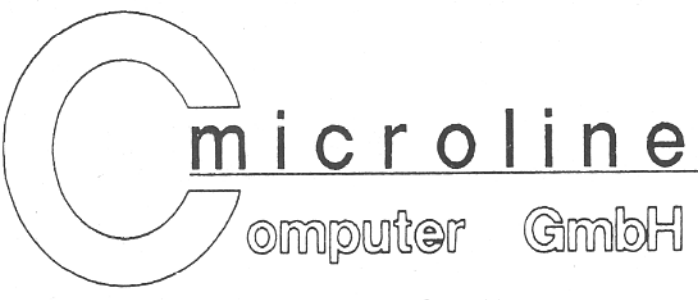
Um aber neue Felder auszuschneiden, muß man sich von dem relativ sicheren Rand entfernen und in Richtung Mitte steuern, da aber lauert der Drache - ein allzu intimer Kontakt sollte nach Möglichkeit vermieden werden. Zwar ist der Spieler am Rand vor dem Drachen sicher, dafür lauern dort einige Totenköpfe.

Bis jetzt mit Qix weitgehend identisch, im Westen also nichts Neues? Falsch, die Gags von Maniax liegen im Detail. Umherfliegende Herzen sorgen für mancherlei Überraschungen. Der Spieler erhält hierdurch eine Lasergun oder einen kleinen Bombenvorrat, um den Totenköpfen endgültig den Hahn abzudrehen und vieles mehr.

Ein weiterer Höhepunkt von Maniax sind die Hintergrundgrafiken des Spielfeldes, die wie ein Puzzle Stück für Stück ausgeschnitten und dadurch sichtbar werden. Es erscheint jeweils ein digitalisiertes Bild einer bekannten Stadt, deren Namen man durch umherfliegende Buchstaben zusammensetzen kann.

Trotz der alten Spielidee ein starkes Spiel für den Atari ST, an dem auch Pazifisten ihre Freude haben! kuw

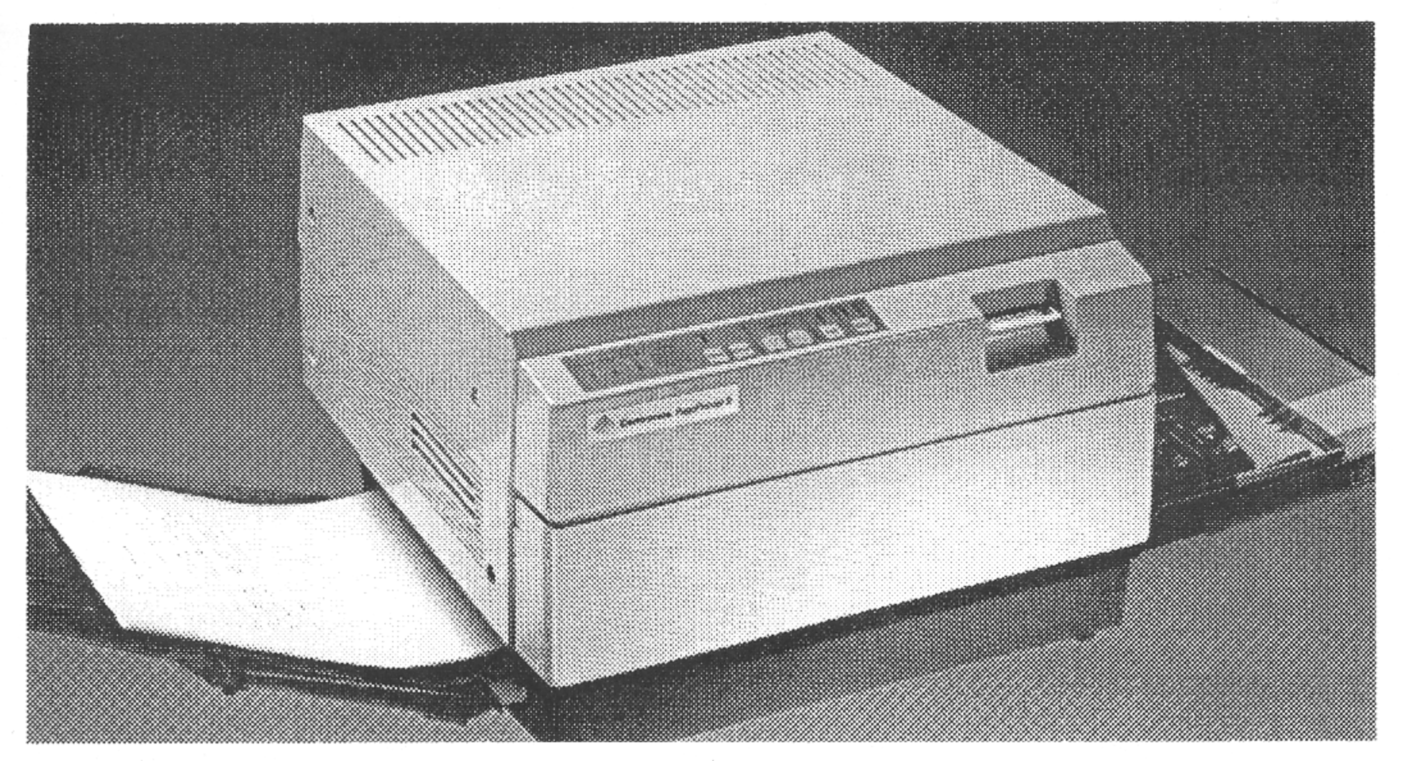




069
626354

microline computer GmbH
Darmstädter Landstr. 33, 6000 Frankfurt 70

CENTRONICS *LASER* PAGEPRINTER 8



Warum träumen Sie nur von einem ***Laserdrucker*** und plagen sich Ihrem alten ***Nadeldrucker*** herum?

Ach, Sie meinen ein Laserdrucker wäre für Sie unerschwinglich? ***Falsch!*** Denn wir machen Ihnen das Umsteigen leicht.

Für nur *DM 4298.-* sind Sie dabei, und zwar mit einem der modernsten Laserdrucker auf dem Markt. Nicht "irgendein" Fabrikat, sondern ein *CENTRONICS LASER PAGEPRINTER 8*.

Warum heißt der Pageprinter 8 ***Pageprinter 8*** ? Ganz einfach: Er schafft mühelos 8 Seiten pro Minute. Und dabei ist ***volle Grafikfähigkeit*** durch ***1.75 Megabyte eingebautes RAM*** ganz selbstverständlich.

Genauso selbstverständlich ist die einfache Anpassung an Ihre Hardware:
Die universelle ***Centronics Schnittstelle*** versteht sich blendend mit Ihrem ***Atari-Computer*** (oder mit jedem anderen Computer), und zahlreiche ***Emulationen*** lassen den ***Pageprinter 8*** für Ihre Software wie einen *HP Laserjet Plus, Diabolo 630, Diabolo 630 ECS, Epson FX-80, IBM Graphics Printer* oder *IBM ProPrinter* erscheinen.

Der einzige Unterschied zwischen dem ***Pageprinter 8*** und dem emulierten Drucker besteht dann in der durch ***300 DPI*** (das entspricht einer Auflösung von ***0.08 mm***) bestechend scharfen Druckqualität. Und mit sogenannten ***Font Modules***, die an der Frontseite einfach eingeschoben werden, bleiben Sie immer auf dem laufenden, was ***neue Zeichensätze*** angeht.

Übrigens wurde die ***ST VISION***, die Sie gerade in den Händen halten, auch mit einem ***Pageprinter 8*** ausgedruckt.

LASER PAGEPRINTER 8 — DM 4298.-

Komplett mit 1.5 Megabyte Speichererweiterung und 5 Druckemulationen

ECHTE PROGRAMMIERER MEIDEN PASCAL

Eine (nicht ganz) ernstgemeinte Studie über die Merkmale von Programmierern der vergangenen Zeiten - Teil 2

Der erste Teil endete mit den unglaublichen Heldentaten der Programmierer von Navigationsprogrammen für Satelliten. Genau an diesem Punkt werden wir im zweiten Teil wieder einsteigen und wünschen den Lesern viel Spaß.

Echte Programmierer

Einige der großartigen Echten Programmierer arbeiten im Jet Propulsion Laboratory in Kalifornien. Viele von ihnen kennen das gesamte Betriebssystem der Pioneer- und Voyager-Sonden auswendig. Mit einer Kombination aus großen, bodengebundenen Fortran-Programmen und kleinen, von den Sonden mitgeführten Assemblerprogrammen vollbringen sie unglaubliche Kunststücke der Navigation und Improvisation.

So treffen sie nur 5 Meilen große Fenster nahe Saturn nach 6 Jahren Flug durch den Weltraum, oder reparieren bzw. umgehen defekte Sensoren, Sender oder Batterien. Angeblich soll es einem Echten Programmierer sogar gelungen sein, in ein paar hundert Byte unbenutzten Speichers innerhalb der Voyager-Sonde ein Mustererkennungsrogramm zu pressen, daß einen neuen Mond des Jupiters suchte, fand und photographierte.

Für die Galileo-Sonde ist vorgesehen, daß sie auf ihrem Weg zum Jupiter entlang einer schwerkraftgelenkten Bahn am Mars vorbeizieht. Diese Bahn führt in einer Entfernung von 80 +/- 3 km an der Marsoberfläche vorbei. Kein Mensch würde diese Art der Navigation einem Pascal-Programm oder -Programmierer anvertrauen.

Viele der Echten Programmierer dieser Welt arbeiten für die amerikanische Regierung, meist für das Verteidigungsministerium. So soll es sein. In letzter Zeit allerdings erscheinen dunkle Wolken am Horizont der Echten Programmierer. Es scheint, als hätten einige einflußreiche Müsli-Fresser im Verteidigungsministerium entschieden, daß in Zukunft alle Verteidigungsprogramme in so einer Art von großer, vereinheitlichter Programmiersprache namens ADA geschrieben werden müßten.

Lange Zeit schien es, als läge ADA's Bestimmung im Verstoß gegen alle Regeln der Echten Programmierung. Es ist eine Sprache mit Strukturen, Datentypen, strenger Typenbindung und Semikolons. Kurz, sie ist wie geschaffen um die Kreativität des typischen Echten Programmierers zu verkrüppeln.

Glücklicherweise hat die jetzt vom DoD ausgewählte Sprache noch genügend interessante Eigenschaften, um dem Echten Programmierer eine Annäherung zu ermöglichen: Sie ist unglaublich komplex, sie enthält Möglichkeiten, um mit dem Betriebssystem herumzujonglieren und Speicherbereiche neu zu verteilen und Edsgar Dijkstra mag sich nicht. Dijkstra ist, wie man wissen sollte, der Autor von "GOTO's Considered Harmful", einem Meilenstein der Programmiermethodologie, der von Pascal-Programmierern und Müsli-Fressern gleichermaßen bewundert wird. Und außerdem, ein zu allem entschlossener Echter Programmierer kann in jeder Sprache Fortran-Programme schreiben.

Der Echte Programmierer kann allerdings auch Kompromisse in Bezug auf seine Prinzipien eingehen und an etwas geringeren Aufgaben als der Vernichtung des Lebens arbeiten, sofern er dafür entsprechend bezahlt wird.

Viele Echte Programmierer schreiben z.B. Videospiele für Atari, allerdings spielen sie nicht damit. Ein Echter Programmierer weiß, wie er die Maschine jedesmal schlagen kann, und damit ist es keine Herausforderung mehr. Jeder bei Lucas-Film ist ein Echter Programmierer, denn es wäre doch verrückt, daß Geld von 50 Millionen Star Wars Fans auszuschlagen.

Der Anteil der Echten Programmierer im Bereich der Computer-Graphics ist etwas niedriger als anderswo, was wahrscheinlich daran liegt, daß noch niemand irgendeinen Nutzen der Computer-Graphics entdeckt hat. Andererseit werden Computer-Graphics überwiegend in Fortran abgehandelt, daher gibt es einige Leute, die

so das Schreiben von Cobol-Programmen vermeiden.

Im allgemeinen spielt der Echte Programmierer wie er arbeitet - mit Computern. Er ist ständig darüber erheitert, daß sein Arbeitgeber ihn tatsächlich für etwas bezahlt, was er nur so zum Spaß sowieso tun würde - allerdings achtet er darauf, diese Meinung nicht zu laut zu äußern. Gelegentlich kommt der Echte Programmierer auch aus seinem Büro heraus, um sich ein wenig frische Luft und ein oder zwei Bierchen zu genehmigen.

Hier daher einge Hinweise, wie man den Echten Programmierer außerhalb des Computerraums erkennt:

- Auf Parties stehen Echte Programmierer in einer Ecke und diskutieren über Sicherheitsmechanismen von Betriebssystemen und wie man darum herum-programmiert.
- Bei Fußballspielen vergleicht der Echte Programmierer die Ergebnisse mit seinen auf grünliniertem Leporello-Papier gedruckten Simulations-ergebnissen.
- Am Strand zeichnet der Echte Programmierer Datenfluß-diagramme in den Sand.
- Ein Echter Programmierer geht in die Disco um sich die Lichtorgel anzusehen.
- Bei Begräbnissen sagt der Echte Programmierer typischerweise "Armer Hans-Helmut, er war mit seinem Sortier-programm schon fast fertig, als ihn der Herzinfarkt erwischt hat".
- Im Supermarkt besteht der Echte Programmierer darauf, seine Bierdosen selber über das Fenster des Strich-codelesers zu schieben, weil er keinem Kassierer zutraut, dies beim ersten Versuch richtig zu machen.

In welcher Umgebung der Echte Programmierer am besten funktioniert? Nun, dies ist eine sehr wichtige Frage für die Manager von Echten Programmierern. Wenn man bedenkt, wie teuer es ist einen von ihnen im Betrieb zu halten, dann sollte man ihn oder sie in eine optimale Arbeitsumgebung versetzen.

Der typische Echte Programmierer lebt vor einem Computerterminal. Rund um dieses Terminal liegen Ausdrucke von jedem Programm, an dem er je gearbeitet hat, sie stapeln sich grob chronologisch geordnet auf jeder ebenen Fläche des Büros.

Im Zimmer verteilt finden sich über ein Dutzend mit kaltem Kaffee mehr oder weniger gefüllte Tassen. Gelegentlich schwimmen Zigarettenkippen darin herum, in einigen Fällen auch Reste von Orangenschalen. Irgendwo liegen Kopien des OS JCL Manuals und der "Principles of Operation" an einer besonders interessanten Stelle aufgeschlagen herum, außer bei extrem guten Leuten.

An der Wand klebt ein Schnelldruckkalender mit Snoopy darauf aus dem Jahr 1969. Über den Boden verteilt liegen Reste der Verpackungen von gefüllten Keksen (der Typ, der schon in der Fabrik so furztrocken gebacken wird, daß er auch bei längerem Liegen im Automaten nicht schlechter wird).

Schließlich, in der linken, oberen Schublade des Schreibtisch, unter der Schachtel mit Muntermachern, liegt eine Schablone für Flußdiagramme, die sein Vorgänger dort vergessen hat. Echte Programmierer schreiben Programme und keine Dokumentation, daß überläßt man den Typen von der Wartung.

Der Echte Programmierer ist in der Lage, 30, 40, ja sogar 50 Stunden in einem Rutsch zu arbeiten, und das unter hohem Zeitdruck. Genaugenommen mag er es so am liebsten. Schlechte Antwortzeiten regen den Echten Programmierer nicht auf - sie geben ihm die Chance, zwischen zwei Kommandos ein bißchen Schlaf zu ergattern.

Wenn die Planung nicht genug Zeitdruck bereithält, dann tendiert der Echte Programmierer dazu, seine Arbeit herausfordernder zu machen, indem er sich die ersten neun Wochen mit einem kleinen, aber sehr interessanten Teil des Problems befaßt, um dann in der letzten Woche seine Aufgabe in zwei oder drei 50-Stunden Marathonsitzungen zu beenden.

Dies beeindruckt nicht nur den Manager, sondern schafft gleichzeitig eine hervorragende Entschuldigung für das Fehlen der Dokumentation.

Und überhaupt: Kein Echter Programmierer arbeitet von 9 bis 5, außer denen von der Nachtschicht. Echte Programmierer tragen keine Schlipse. Echte Programmierer tragen keine hochhackigen Schuhe. Echte Programmierer kommen zur Arbeit, wenn andere zum Mittagessen gehen. Ein Echter Programmierer vergißt vielleicht den Vornamen seiner Angetrauten, aber niemals den Inhalt der gesamten ASCII-Tabelle. Echte Programmierer können nicht kochen. Da Supermärkte um 3 Uhr morgens selten geöffnet sind, müssen sie sowieso von Kaffee und Keksen leben.

Die Zukunft betrachtend machen sich eine Reihe von Echten Programmierern Sorgen, daß die jüngste Programmierergeneration nicht mehr mit der gleichen Lebensperspektive aufwächst, wie sie selbst. Viele der Jüngeren haben noch nie einen Computer mit einer Maschinenkonsole gesehen. Kaum ein Schulabgänger kann heute noch hexadizimal rechnen, ohne einen Taschenrechner zu benutzen.

Die Studenten von heute sind weich - geschützt von den Realitäten der Programmierung durch symbolische Debugger, Texteditoren die Klammern zählen und benutzerfreundliche Betriebssysteme. Und das Schlimmste ist, einige dieser angeblichen Computerspezialisten kommen zu Rang und Namen, ohne je Fortran zu

lernen! Sind wir dazu verdammt, eine Industrie von UNIXHackern und Pascal-Programmierern zu werden?

Nun, aus meiner Erfahrung heraus glaube ich behaupten zu dürfen, daß das Schicksal den Echten Programmierern wohlgesonnen ist. Weder OS/370 noch Fortran zeigen irgendwelche Symptome des Aussterbens, trotz aller Anstrengungen der Pascal-Programmierer. Selbst subtilere Tricks wie das Hizufügen strukturierter Schleifen zu Fortran sind fehlgeschlagen.

Sicher, einige Computerhersteller liefern Fortran 77 Compiler, aber jeder einzelne von ihnen läßt sich über eine einzige Compiler-Option in einen Fortran 66 Compiler verwandeln - mit DO Schleifen, wie von Gott geschaffen.

Selbst UNIX scheint für den Echten Programmierer nicht mehr so schlecht zu sein wie früher. Die neueste UNIX-Version hat das Potential eines Betriebssystems, das eines Echten Programmierers würdig ist. Sie hat zwei verschiedene, leicht inkompatible Benutzer-Schnittstellen, einen geheimnisvollen und komplizierten Teletype-Treiber und virtuellen Speicher.

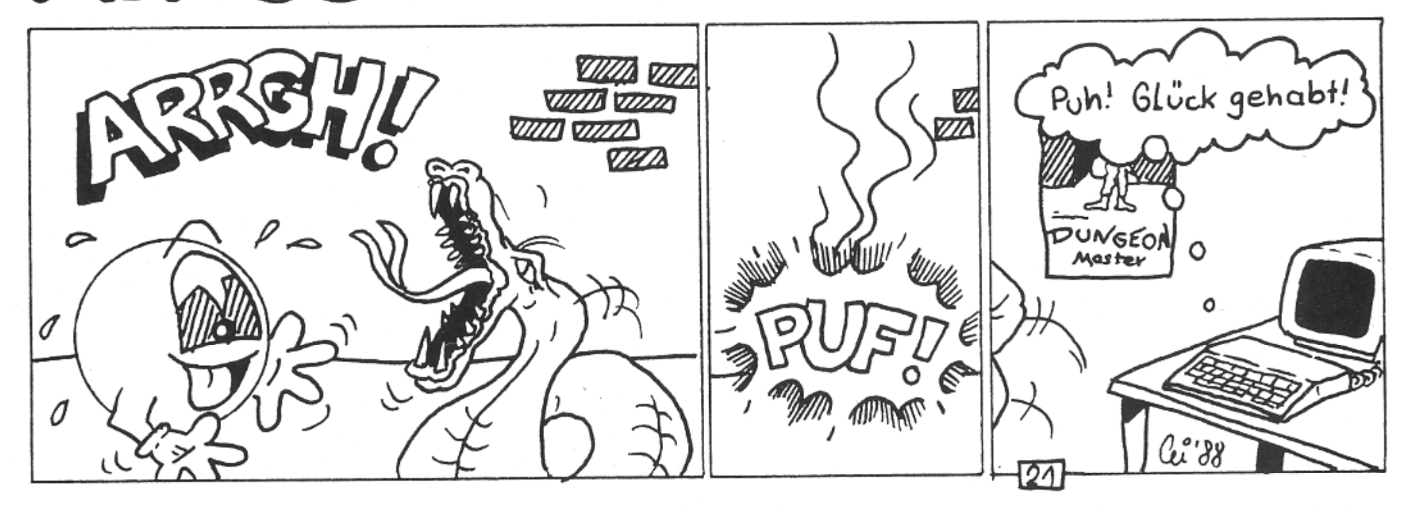
Und wenn der Echte Programmierer die Strukturierung ignoriert, dann er sich sogar mit C anfreunden. Schließlich gibt es keine Typenbindung, Bezeichner sind sieben Zeichen lang, und man hat Zeiger als Bonus. Das ist, als hätte man die besten Teile von Fortran und Assembler vereint, von den kreativeren Möglichkeiten des "#define" ganz zu schweigen.

Nein, die Zukunft ist nicht völlig schlecht. So hat sich in den vergangenen Jahren die populäre Presse sogar über die clevere neue Brut von Computer-Hackern geäußert, die Plätze wie Standford oder das MIT zugunsten der Wirklichkeit verlassen haben.

Allen Anzeichen nach lebt der Geist der Echten Programmierer weiter in diesen jungen Männern und Frauen. Und solange es schlecht beschriebene Ziele, bizarre Fehler und unrealistische Zeitpläne gibt, solange wird es Echte Programmierer geben, die bereit sind einzuspringen und das Problem zu lösen und die sich die Dokumentaion für später aufheben.

Lang lebe Fortran !

Aus Datamation 7/83, von Ed Post, übersetzt von Bernd Leeds.



ST-MATH
Die Programmiersprache für mathematische Anwendungen !?!

Zuerst einmal die Autoren von ST-MATH: Jörg Begemann & Alexander Niemeyer.

ST-MATH ist ein professionelles Programm für symbolische Algebra und Analysis. Geschrieben wurde es vollständig in Assembler und erzielt damit eine extreme Geschwindigkeit. Es ist zudem ein TOS-Programm, welches den Aufbau einer LISP ähnlichen Programmiersprache hat.

Der Aufbau

Die äußere Form von ST-MATH ist der von muMATH nachempfunden. Damit können die bestehenden Programme dieser Sprache weiter verwendet werden.

Der interne Aufbau ist dagegen verschieden. Das Programm muMATH ist ein interpretiertes Hochsprachenprogramm, ST-MATH dagegen, wie schon gesagt, völlig in Assembler geschrieben.

Rundungsfehler!?!

Damit bei ST-MATH keine Rundungsfehler entstehen, arbeitet es nur mit ganzen Zahlen. Dies ist aber kein Problem, da ST-MATH die Bruchrechnung perfekt beherrscht.
Als sogenannten Bonus gibt es auch noch eine Rechengenauigkeit von '610' Stellen. Wo gibt es so etwas sonst noch, wenn man bedenkt, daß die meisten Sprachen 'nur' bis zu ein paar hundert Stellen arbeiten. Dies mit einer Genauigkeit, die zwischen 9 und 19 Stellen schwankt. Dazu kann man sich noch ein Zahlensystem zwischen 2 und 36 aussuchen, welches natürlich ständig geändert werden kann.

170 Befehle!!!

Um die Komplexität von ST-MATH aufzuzeigen, sei erwähnt, daß über 170 Befehle implementiert sind. Wobei man hier wirklich nicht sagen kann: 'Lieber Qualität als die Quantität der Masse.' Denn hier ist alles Qualität. Und wem dies noch nicht genügt, der kann auch noch seine eigenen Assemblerprogramme in die Programme von ST-MATH einbinden.

Programmaufbau

Gehen wir jetzt ein wenig näher auf ST-MATH ein:
Das Programm ist, wie schon gesagt, LISP ähnlich aufgebaut und erlaubt somit beliebig lange Eingaben, die nur vom Speicherplatz begrenzt werden. Da aber eine Zeile nicht endlos lang ist, können die Eingaben über beliebig viele Zeilen verteilt werden. Damit ST-MATH erkennt, wann eine Eingabe beendet ist, muß an den Schluß ein Terminator gestellt werden. Von diesen Terminatoren gibt es drei verschiedene.

Terminatoren

Der Erste ist das Semikolon (';'). Es veranlaßt, daß die Eingabe ausgewertet wird und das Ergebnis in normaler mathematischer Darstellung ausgegeben wird.

Der Zweite ist das Dollarzeichen ('$'). Es bewirkt ebenfalls, wie alle Terminatoren, die Auswertung der Eingabe. Das Ergebnis wird aber nicht auf dem Bildschirm angezeigt.

Der dritte und letzte Terminator ist das Ampersand ('&'). Dieser Terminator veranlaßt eine Ausgabe der Auswertung der Eingabe in einer listenförmigen Notation. Dies hat damit nur eine Auswirkung bei Listen. Die Liste wird in den first- und rest-Teil aufgespalten. Um das zu verdeutlichen, gebe ich ihnen am besten ein kleines Beispiel: Eine ineinander verschachtelte Liste soll in der listenförmigen Notation ausgegeben werden. Die Liste lautet:
(1, 2, 3, (4, 5, 6), 7, (8), 9)

Die Ausgabe mit dem Terminator '&' würde dann wie folgt aussehen:
1 (2, 3, 4(5, 6), 7, 8(), 9)

Variable

In der Systemvariablen '@', dem 'Klammeraffen', ist immer das letzte Ergebnis enthalten. Um aber noch mehr Werte zwischenspeichern zu können, gibt es die Variablen. Diese müssen mit einem Buchstaben beginnen und können ansonsten aus Buchstaben und Zahlen bestehen. Hierbei ist nur zu berücksichtigen, daß die Länge des Variablennamens '254 Zeichen' nicht überschreitet. Den Variablen können beliebige Ausdrücke zugewiesen werden (z.B. Zahlen, Namen, Matrizen, etc.).

Die Befehle

Doch genug der Vorrede. Was bietet ST-MATH, daß es zu so einem leistungsstarken Programm wird?

Um klein zu beginnen:
ST-MATH hat Funktionen um Tasten mit beliebig langen Befehlskombinationen zu belegen und abzufragen.
Es hat eine Reihe von Konstanten (z.B. Pi, e, (-1)^(1/2)=Wurzel aus -1, PINF=Plus unendlich, etc.).
Es besitzt Kontrollvariablen (14 Stück), die die Ausgabe von

mathematischen Termen steuern (z.B. sind diese beiden Ausdrücke gleich A^(B+C), A^B+A^C). Diese Kontrollvariablen sind sehr wichtig für den Benutzer. Denn mit ihnen kann er Ergebnisse nach seinem Willen ausgeben. Durch Ausprobieren der diversen Schreibweisen kann man selbst da noch zu einem Ergebnis kommen, wo alle Logik ein Ende hat.

Auch die Ermittlung von dem größten gemeinsamen Teiler und des kleinsten gemeinsamen Vielfachen ist durch einfache Befehle realisiert worden.

Komplexe Befehle

Jetzt wird es etwas komplizierter:
Die Differentialrechnung wird von ST-MATH aus dem 'eff eff' beherscht. So kann man nach jeder Variablen in einem beliebigen Term ableiten. Dabei spielt es keine Rolle, ob es ein trigonometrischer Term ist, oder wie komplex er auch immer sein mag. Dies ist z.B. gut geeignet für die Kurvendiskussion. Es können die Extremwertstellen und die Nullstellen errechnet werden, wobei diese Befehle beliebig ineinander verschachtelt werden können.

Auch das Thema Gleichungen ist für ST-MATH kein Problem. Man kann mit ihnen rechnen, Variablen Werte zuweisen und sich das Ergebnis ansehen, sie vereinfachen, usw..

Die Grenzwertberechnung ist ebenso unproblematisch für ihn. Hierfür gibt es den Befehl LIM(a,b,c,d). Dieser ermittelt den Grenzwert des Terms a zur Variable b, mit der zu untersuchenden Stelle c und der Richtung d.

Als nächstes kommen wir zur Integralrechnung. Hierzu gibt es die Befehle INT(a,b) und DEFINT(a,b,c,d). Diese dienen zum einen zur Integration und zum anderen zur Berechnung von Flächeninhalten unter Kurven.

Ein sehr interesanter Bereich dürfte sicherlich die Matrizen- und Vektorrechnung sein. Er unterstützt dabei den Zeilen- und Spaltenvektor (und dessen Kombinationen). Vektoren können beliebig ineinander Verschachtelt werden. Es können alle Rechenoperationen angewandt werden. Zusätzlich gibt es spezielle Befehle für die Matrizen. Einmal zur Abfrage von Elementen in Matrizen, dann zur Erzeugung von einer Einheitsmatrix, zum Transponieren einer Matrix. Es gibt Befehle zur Multiplikation und Division von Matrizen und natürlich kann auch die Determinante bestimmt werden.

Es sind auch Befehle zur Summen- und Produktberechung implementiert. Da wäre zum einen der Befehl SIGMA(a,b,c,d). Dieser berechnet die Summe des Terms a, wobei b die Laufvariable ist. Die Summe wird von c bis d errechnet. Zum anderen gibt es den Befehl PROD(a,b,c,d). Dieser ist analog zum SIGMA-Befehl, dabei wird jetzt allerdings das Produkt bestimmt.

Die Taylor-Reihen ermöglichen es einem, eine beliebige Funktion durch die Entwicklung in Potenzreihen näherungsweise zu bestimmen. Der Befehl TAYLOR(a,b,c,d) entwickelt dabei die Reihe und der Befehl EVSUB(a,b,c) wertet diese aus.

Trigonometrische Befehle sind ebenfals zahlreich vorhanden. Da wären z.B. der sin, cos, csc, cot, asin, atan, etc..

Zur logischen Verknüpfung stehen nicht nur die wichtigsten Befehle zur Verfügung, nein es sind auch einige hilfreiche Zusatzbefehle enthalten. So gibt es z.B. die Befehle AND, OR, EQ (EQ testet auf absolute Gleichheit), etc..

Es sind Testfunktionen implementiert, mit denen man Daten nach bestimmten Kriterien überprüfen kann (z.B. ob Zahl, ob Positiv, ob Name, etc.).

Die Sprache

Dies waren erst ein mal ein paar Befehle von ST-MATH, aber ST-MATH wäre keine Sprache, wenn man sie nicht programmieren könnte.

Da sie LISP orientiert ist, arbeitet sie mit den folgenden drei Datenstrukturen:
Namen, Zahlen, Knoten
Diese bestehen alle aus Zeigern, was sich günstig auf die Arbeitsgeschwindigkeit, den Speicherplatz und die Handhabung auswirkt.

Datenstrukturen

Namen haben drei Zeiger. Wobei der erste auf den Wert des Namens, der zweite auf die Propertyliste und der dritte auf die Funktionsdefinition (falls diese existiert) zeigt.

Zahlen haben ebenfalls drei Zeiger. Der erste zeigt wiederum auf den Wert der Zahl (also auf sich selbst), der zweite zeigt auf das Vorzeichen der Zahl und der dritte zeigt auf den Speicherbereich, in dem die Zahl abgelegt ist.

Knoten bestehen aus zwei Zeigern. Sie dienen zur Verkettung von Listen. Der erste Zeiger zeigt dabei auf ein Element und der zweite auf den nächsten Knoten.

Diese Datenstrukturen können auch ausgetauscht, aufgespalten, verschmolzen und verbunden werden.

Die Routinen

Um 'Programme' zu schreiben, gliedert man diese in Funktionen und Subroutinen. Diese werden wiederum durch When-Exit, Block und Loop strukturiert.

Wie sieht nun die Syntax einer Funktion oder Subroutine aus? Sie fängt mit dem Befehl, dem Namen der Funktion und den Parametern an. Danach besteht sie aus Anweisungen die in jeder Zeile mit einem Komma

abgeschlossen werden. Zum Schluß wird sie noch mit einem Endbefehl abgeschlossen.

Das Interessanteste daran ist die Parameterübergabe. Bei ihr gibt es zwei verschiedene Arten. Als erstes die spread- und als zweites die nospread-Funktion. Bei einer spread-Funktion stehen die Parameter in einer Klammer hinter dem Funktionsnamen. Weiß man nicht wieviele Parameter man übergeben will, so verwendet man die nospread-Funktion. Bei ihr steht 'ein' Parameter 'ohne' Klammer hinter dem Programmnamen. In diesen Parameter werden alle übergebenen in eine Liste geschrieben. In welcher anderen Programmiersprache hat man so eine Möglichkeit!

Der Unterschied zwischen einer Funktion und einer Subroutine ist der, daß bei einer Funktion die Parameter vor der Übergabe ausgewertet werden, bei einer Subroutine nicht.

Zur Verdeutlichung:
Die Variable Test1 hat den Wert von 11. Wird nun diese zu einer Funktion übergeben, kommt bei der Funktion der Wert 11 an. Wird aber die Variable an eine Subroutine übergeben, so kommt der Name Test1 an.

Programmstrukturen

Die drei Programmstrukturen dürften jedem eigentlich schon bekannt sein. Hier aber noch einmal eine kurze Beschreibung. Die Struktur 'WHEN vergleich, anweisung EXIT, anweisung' führt einen Vergleich aus. Ist dieser true, so wird die Anweisung vor Exit, ansonsten hinter Exit ausgeführt. Um aber zu entscheiden, wann der Befehl hinter Exit fertig ist, wurde die Struktur Block erschaffen. Dieser wird linear durchgearbeitet, bis der Parser auf einen When-Befehl stößt. Ist dieser true, so wird nun von Exit zu Endblock gesprungen. Der Loop-Befehl wird solange wiederholt, bis mit einem When-Befehl aus dieser Struktur ausgestiegen wird.

Die Ausgabe

Doch was helfen uns diese ganzen Strukturen, wenn es keine Ausgabebefehle gibt.

Dazu gibt es zum einen Befehle zur Ausgabe auf dem Bildschirm und zum anderen zur Ausgabe auf einem Laufwerk. Will man ein geschriebenes Programm abspeichern, so wird, da ST-MATH eine variable Speicherverwaltung besitzt, der gesamte benutzte Speicher als SYS-File abgespeichert.

Die Rekursion ist auch dabei!

Aber jetzt gehts in die Vollen!
Um ST-MATH zu einem komplexen und mächtigen Werkzeug zu machen, beherrscht dieses auch die Rekursion. Dies bedeutet, daß es in ST-MATH lokale Variable gibt, daß Funktionen von sich aus aufgerufen werden und, daß Werte zurückgegeben werden können.

Dies sind eine Auswahl von Befehlen, die ST-MATH beherrscht. Der Interpreter ist dermaßen leistungsstark, daß ich jedem, der des öfteren mathematische Probleme lösen muß, empfehlen kann, mit dieser Sprache zu arbeiten.

Das Handbuch

Zum Handbuch muß auch noch etwas gesagt werden. Dies ist mehr eine gutgelungene Einführung in die Sprache, als nur ein einfaches Handbuch. Wobei das eigentliche Handbuch aber leider dermaßen zu kurz kommt, daß es weder ein alphabetisches Verzeichnis aller Befehle gibt, noch einen Index oder ein Sachregister. Dies macht es nahezu unmöglich mit diesem Buch zu arbeiten. Sucht man z.B. einen bestimmten Befehl so muß man das ganze Buch durchblättern, da man nirgendwo einen Tip findet, der einem das Suchen erleichtert. Und dies in einem 230 Seiten starken Wälzer. Noch nicht einmal im Inhaltsverzeichnis stehen die Seitenzahlen.

Ich habe mich also als erstes in Tagelanger Arbeit daran gemacht ein Quickreferenzmanual zu ST-MATH zu erstellen, um überhaupt mit dieser Supersprache arbeiten zu können. Diese ist 19 Seiten lang und enthält alle Befehle in alphabetischer Reihenfolge mit Erklärung, dazu eine Befehlsübersicht mit allen Befehlen und deren Vorkommen im Buch und in meinem Manual.

Zu guter letzt habe ich das Inhaltsverzeichnis des Buchs überarbeitet und die einzelnen Kapitel und Unterkapitel mit Seitenzahlen versehen. Mit diesem Zusatz ist es mir nun endlich möglich, 'so richtig gemütlich' in ST-MATH zu programmieren.

ST-MATH gibt es beim Heimverlag für 98.-- DM

Anbei noch ein Auszug aus der oben beschriebenen Befehlsübersicht.

mas

Befehl	Art
ACOS(a)	Trigonometrie
ACOT(a)	Trigonometrie
ACSC(a)	Trigonometrie
ADJOIN(a,b)	Konstruktionsfunktion
a AND b	Rechenregel
APPLY(a,b)	Auswertungsfunktion
#ARB	Konstante
ASEC(a)	Trigonometrie
ASIN(a)	Trigonometrie
ASSIGN(a,b)	Zuweisungsfunktion
ASSOC(a,b)	Propertyfunktion
ATAN(a)	Trigonometrie
ATOM(a)	Testfunktion
BASEXP	Kontrollvariable
BLOCK	Programmstruktur
CINF	Konstante
COMPRESS(a)	Subatomische Funktion
CONCATEN (a,b)	Modifizierungs Funktion
COS(a)	Trigonometrie
COT(a)	Trigonometrie
CSC(a)	Trigonometrie
DEFINT(a,b,c,d)	Integralrechnung

Befehl	Art
DELIMITER	Parserfunktion
DENDEN	Kontroll-variablen
DENNUM	Kontroll-variablen
DET(a)	Matrizen-rechnung
DIF(a,b)	Differential-rechnung
DIFFERENCE (a,b)	Arithmetische-funktion
DIVIDE(a,b)	Arithmetische-funktion
#E	Konstante
ECHO	Printerkontroll-variable
ECHO()	Printerkontroll-variable.
EMPTY(a)	Testfunktion
a EQ b	Rechenregel
ESC	Systemtaste
EVAL(a)	Funktions-auswertung
EVEN(a)	Testfunktion
EVSUB(a,b,c)	Auswertungs-funktion
EXPAND(a)	Kontroll-variablen
EXPBAS	Kontroll-variablen
EXPD(a)	Kontroll-variablen
EXPLODE(a)	Subatomische-Funktion
FALSE	Konstante
FCTR(a)	Kontroll-funktion
FIRST(a)	Selektions-funktion
FLAGS()	Systemfunktion
FUNCTION	Funktions-definition
GCD(a,b)	Arithmetische-Funktion
GET(a,b)	Property-Funktion
GETD(a)	Funktions-abfrage
GREATER(a,b)	Vergleichs-funktion
#I	Konstante
IDMAT(a)	Matrizen-rechnung
INFIX	Propertyliste
INT(a,b)	Integral-rechnung
INTEGER(a)	Testfunktion
KEY(a,b,c)	Systemfunktion
KEY(a)	Systemfunktion
LBP	Propertyliste

Befehl	Art
LCM(a,b)	Arithmetische-Funktion
LENGTH(a)	Subatomische-F.
LESSER(a,b)	Vergleichs-funktion
LINELENGTH (a)	Systemfunktion
#LIM	Konstante
LIM(a,b,c,d)	Grenzwert-berechnung
LIST(a,b,...)	Terminator
LN(a)	Logarithmische-Funktion
LOAD(a,b)	Ladefunktion
LOG(a,b)	Logarithmische-Funktion
LOGBAS	Kontroll-variablen
LOGEXPD	Kontroll-variablen
LOOP	Programm-struktur
MEMBER(a,b)	Vergleichs-funktion
MINF	Konstante
MINUS(a)	Arithmetische-Funktion
MOD(a,b)	Arithmetische-Funktion
MOVD(a,b)	Funktions-zuweisung
MZERO	Konstante
NAME(a)	Testfunktion
NEGATIVE(a)	Testfunktion
NEWLINE	Systemfunktion
NEWLINE(a)	Systemfunktion
NOT(a)	Modifizierungs-funktion
NUMDEN	Kontrollvariable
NUMNUM	Kontrollvariable
OBLIST()	Systemfunktion
a OR b	Rechenregel
ORDERP(a,b)	Vergleichs-funktion
PBRCH	Kontrollvariable
#PI	Konstante
PINF	Konstante
PLUS(a,b)	Rechenfunktion
POP(a)	Zuweisungs-funktion
POS()	Systemfunktion
POSITIVE(a)	Testfunktion
PREFIX	Propertyliste
PRINT(a)	Printfunktion
PRINTLINE	Kontrollvariable
PRINTLINE(a)	Printfunktion
PROD(a,b,c,d)	Produkt-berechnung
PRTMATH (a,b,c,d)	Printfunktion

Befehl	Art
PUSH(a,b)	Zuweisungs-funktion
PUT(a,b,c)	Property-funktion
PUTD(a,b)	Assembler-einbindung
PWREXPD	Kontrollvariable
PZERO	Konstante
QUOTIENT(a,b)	Arithmetische-Funktion
RADIX(a)	Arithmetische-Funktion
RBP	Propertyliste
RDS(a,b,c)	Readerfunktion
READ	Kontrollvariable
READCHAR	Kontrollvariable
READCHAR()	Readerfunktion
RECLAIM()	Speicherverw.-Funktion
REPLACEF(a,b)	Modifizierungs-funktion
REPLACER(a,b)	Modifizierungs-funktion
REST(a)	Selektions-funktion
REVERSE(a)	Konstruktions-funktion
REVERSE(a,b)	Konstruktions-funktion
RREST(a)	Selektions-funktion
RRREST(a)	Selektions-funktion
SAVE(a,b)	Speicher-funktion
SCAN	Systemvariable
SCAN()	Readerfunktion
SEC(a)	Trigonomet.-Funktion
SECOND(a)	Selektions-funktion
SIGMA(a,b,c,d)	Summen-funktion
SIN(a)	Trigonomet.-Funktion
SOLVE(a,b)	Differential-funktion
SPACES(a)	Systemfunktion
SUBROUTINE	Funktions-definition
SYSTEM()	Systemfunktion
TAN(a)	Trigonomet.-Funktion

Dies war eine Auswahl der Befehle von ST-MATH. Man erkennt sicherlich sofort, daß dieses mächtige Befehle sind, die eine gutausgestattete Sprache ausmachen.

mas

OBLITERATOR

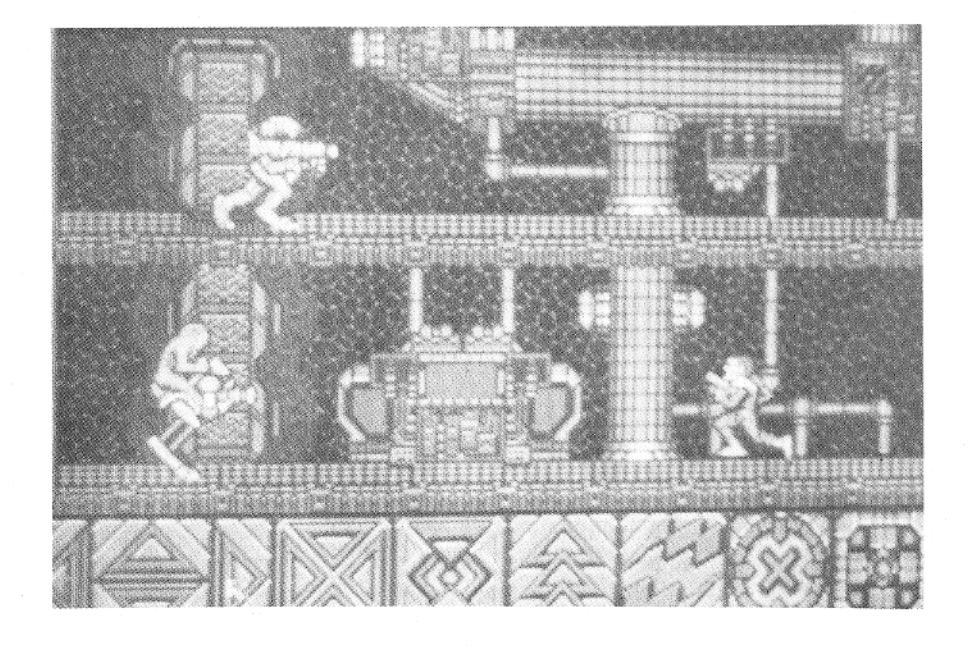
Barbarian im Weltraum

Obliterator, Psygnosis Ltd., England
US$ 40, ca. DM 60-80

Das englische Softwarehaus Psygnosis ist wohl jedem ST Besitzer ein Begriff. Seit dem Beginn der ST Ära im Sommer 1985 versorgte es die ST Gemeinde mit einer Reihe überdurchschnittlicher Spiele, die sich meist durch exellente Grafik auszeichneten. Dies ist auch bei seinem neusten Produkt 'Obliterator' nicht anders.

Erinnern Sie sich noch an Hegor aus dem letzten Psygnosis Hit 'Barbarian' ? Wenn Sie ihm seine goldenen Locken abschneiden, seinen Lendenschurz gegen einen roten Weltraumanzug austauschen und ihn anstelle eines Schwertes eine Laserpistole bzw. Bazooka in die Hand drücken, so erhalten Sie Drak, den Obliterator. Er ist der letzte, der Elite Kampf Einheit der Obliterators. Einer Einheit die speziell für solche Himmelfahrtkommandos trainiert wurde, bei denen die Chancen bestenfalls ungünstig standen. Und wieder einmal ist es Draks Aufgabe die gute alte Mutter Erde zu retten.

Die gesamte irdische Sternenflotte ist von einem einzigen feindlichen Raumschiff ausgelöscht worden und dieses Raumschiff zu vernichten, ist Draks Aufgabe. Er wird an Bord desselben gebeamt, soll dort die Triebwerke, die Waffensysteme und den Schutzschirm durch Entfernen wichtiger Komponenten deaktivieren und dann so schnell wie möglich vor der Zerstörung das Schiff mit Hilfe eines Shuttles verlassen.



Klingt doch alles recht einfach, oder ? Ja, dummerweise ist das Raumschiff aber von oben bis unten vollgepackt mit Außerirdischen aller Arten, Kampfrobotern und Laserfallen in den verschiedensten Ausführungen.

Hier haben sich die Entwickler des Spiels wirklich große Mühe gegeben, damit keine Langeweile aufkommt. Da gibt es grüne echsenartige Wesen, die Drak manchmal zu Fuß verfolgen. Manchmal reiten Sie auf einer Art mechanischem Kängeruh oder auf einem Gefährt, das an die Flugmotorräder aus dem Star Wars Film 'Rückkehr der Jedi Ritter' erinnert. Desweiteren gibt es noch braune Alien-artige Monster, die zuweilen an der Decke entlangfahren.

An anderer Stelle wiederum fallen grüne ballonartige Gebilde auf den Boden und verwandeln sich dort in kleine äußerst flinke zweibeinige Roboter, die an Spielbergs Kinderroboter aus 'Wunder in der 8. Straße' erinnern. Man sieht also, daß die Autoren sich an einer Reihe von Kinofilmen orientiert haben.

Praktisch hinter jeder Tür und jeder Ecke lauert eine neue Überraschung auf unseren Helden. Vorsicht ist auch bei solchen Räumen geboten, die auf den ersten Blick keine Gefahr offenbaren. Meist öffnet sich , sobald sich Drak in Bewegung setzt, irgendeine Geheimtür und Laserkanonen nehmen ihn von dort ins Kreuzfeuer.

Während seiner Mission macht Drak noch Bekanntschaft mit einer ganzen Reihe anderer Roboter und Wesen, die ich hier

aber nicht näher beschreiben möchte, um nicht die ganze Spannung zu nehmen.

Als Waffen stehen Drak eine Laserpistole, ein Lasergewehr, ein Blaster und eine Bazooka zur Verfügung. Am Anfang des Spieles ist er nur mit der Pistole ausgerüstet. Die anderen Waffen muß er sich erst im Laufe des Spieles besorgen - ebenso wie die jeweils entsprechende Munition !!!

Damit der erste Feindkontakt nicht auch zugleich der letzte ist, verfügt Drak natürlich über einen Schutzschirm, der aber um längeres Spielen zu ermöglichen, gelegentlich aufgeladen werden sollte.

Obwohl im Handbuch steht, daß Drak alles und jeden über den Haufen schießen soll, empfiehlt es sich manchmal aber auch die als Verteidigungsbewegung gekennzeichnete Funktion anzuwenden. Befindet sich Drak nämlich in Bewegung, so legt er dann sofort eine gekonnte Hechtrolle hin, die ihn an so manchem unangenehmen Hindernis mit Leichtigkeit vorbeiführt. Eine Maßnahme die in einigen Situationen für Drak gesünder sein könnte, als der Versuch den Angreifer durch einen gezielten (?!) Schuß auszuschalten.

Praktisch ist auch, daß man an bestimmten Stellen die Möglichkeit erhält, den jeweiligen Spielstand abzuspeichern. Bei den unzähligen Gefahren, die im Raumschiff lauern, sollte man dies auf jeden Fall tun.

Das Obliterator sehr stark an Barbarian angelehnt ist, erkennt man bereits am Bildschirmaufbau. Der untere Teil des Bildschirms besteht aus einer Funktionsleiste, mit der man Draks Bewegungen steuern kann. Durch Drücken der Space Taste kann dort wahlweise auch die Bewaffnung und der Zustand seines Schutzschirms angezeigt werden.

Im Gegensatz zu Barbarian, wo der Held sein Leben mit einem markerschütternden digitalisierten Schrei aushauchte, stirbt der letzte Obliterator ungemein nervenschonender, wenn es ihn auch zuweilen buchstäblich in meherer Stücke zerreißt. Die als Ersatz für den digitalisierten Sound ständig im Hintergrund laufende Melodie wird der eine oder andere bestimmt nach ein paar Spielen ausblenden.

Wie schon von anderen Psygnosis Spielen gewohnt, kommt Obliterator in der typischen schwarzen Pappbox. Außer dem Handbuch, den beiden Disketten und der Registration-Card erhält der Käufer auch wieder ein kleines Poster. Dieses zeigt ein Wesens, das der Alienmutter aus dem zweiten Alienfilm in gewisser Weise ähnlich sieht.

Das englische Handbuch umfaßt zehn Seiten. Auf dem Cover wird vor Viren gewarnt und zum Ausschalten des Computers vor Spielbeginn geraten. Im Inneren wird dann neben den Ladeinstruktionen und der Hintergrundstory ausführlich die Spielsteuerung erläutert. Dazu ist zu sagen, daß man Drak per Maus, Joystick und/oder Tastatur steuern kann. Für gezieltes Schießen empfehle ich aber insbesondere die Maus.

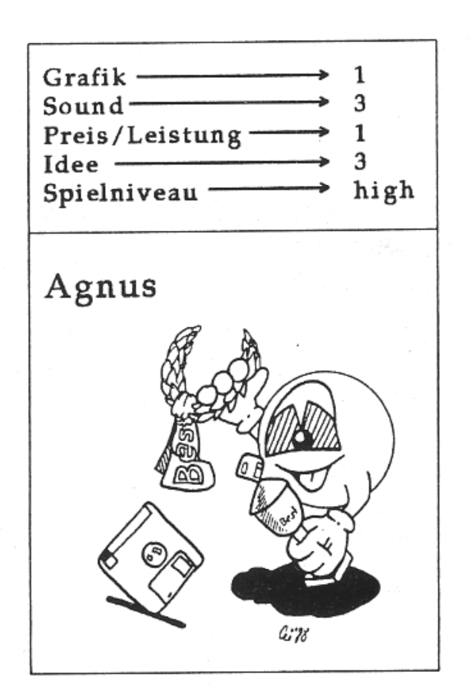
Eine andere für Psygnosis Produkte typische Eigenart trifft auch bei Obliterator leider zu - nämlich das Verwenden eines äußerst aufwendigen Kopierschutzes. Beim Testen des Programmes auf verschiedenen Rechnern, besser - verschiedenen Diskettenlaufwerken traten bei einigen Modellen Ladeprobleme auf. Psygnosis war sich dessen wohl vollends bewußt. Denn sogar in der Anleitung heißt es (frei übersetzt):
"Sollte das Titelbild 45 Sekunden nach dem Start des Bootvorgangs nicht erschienen sein, so handelt es sich entweder um einen Fehler ihres Systems oder der Diskette. Überprüfen Sie ihr System und wiederholen Sie den Ladevorgang. Sollte ihr System in Ordnung sein und ein erneuter Ladeversuch schlägt wieder fehl, so bringen Sie die Programmdisketten zu ihrem Softwareverkäufer zwecks Umtausch."

Eine äußerst fragwürdige Art das Problem der Raubkopien zu lösen, die den sonst recht guten Gesamteindruck des Spieles etwas trübt.

Sieht man einmal von diesem Makel ab, so ist Obliterator eigentlich nur zu empfehlen. Es reiht sich wirklich nahtlos in die Reihe der guten Psygnosis Programme ein. Wenn auch die Animation manchmal nicht ganz so fließend ist, so ist die Grafik im Gegensatz zu dem, was einem in anderen ST Spielen schon geboten wurde, wieder eine Klasse für sich.

Der Preis für Obliterator wird hierzulande bei DM 60-80 liegen, was durchaus im Rahmen liegt. Ich bin sicher das Obliterator auch hier in Deutschland viele Freunde finden wird. Wenn auch die Spielidee nicht gerade die neuste ist, so ist die Ausführung doch recht gelungen. Obliterator wird dem Käufer bestimmt eine Vielzahl an aufregenden Stunden vor dem Computer bescheren.

mts



Grafik →	1
Sound →	3
Preis/Leistung →	1
Idee →	3
Spielniveau →	high

ASSEMBLER PROGRAMMIERUNG

Der zweite Teil dieser Reihe beschäftigt sich mit den Betriebssystemroutinen des Atari ST

Der Prozessor

Das Herz eines jeden Computers ist die CPU (= Central Processing Unit) oder auch Prozessor genannt. Dieser Baustein ist der Motor des Computers und steuert alle wichtigen Funktionen. Um sinnvoll zu Arbeiten benötigt die CPU Anweisungen in Form von binär verschlüsselten Befehlen, die wiederum vom Menschen erstellt werden. Diese Daten, eher bekannt unter dem Namen 'Programm', werden von der CPU während eines Programmlaufes interpretiert. Entsprechend ihrer Bedeutung sendet der Prozessor Signale an angeschlossene Bausteine oder liest Signale aus, um sie weiterzuverarbeiten. Die angeschlossenen Bausteine sind z.B. I/O-Bausteine zur Ansteuerung des Druckers und der Diskettenstation oder die MMU (memory managment unit). Dieser Baustein verwaltet den Hauptspeicher des Computers, bei rund 8 Millionen Bits (= 1 Megabyte) wie sie in einem Atari ST 1040 vorhanden sind, keine leichte Aufgabe.

Ohne die vom Menschen geschriebenen Programme wäre der Prozessor nur ein nutzloses Stück Silikon, dementsprechend wichtig ist der richtige Einsatz der geistigen Kapazitäten zur Programmentwicklung. Nur leider nützt einem seine ganze Intelligenz nichts, wenn man sich nicht über den Aufbau des jeweiligen Prozessors informiert, der im heimischen Rechner eingesetzt wird. Es genügt nämlich nicht den Texteditor zu Laden und einen Brief etwa in der folgenden Form aufzusetzen:

Sehr geehrter MC68000,
hiermit möchten wir Sie erinnern, den unter dem Namen 'LETTER.DOC' gespeicherten Brief endlich auf dem Drucker erscheinen zu lassen !

Wenn es so einfach wäre, gäbe es sicherlich mehr Programmierer. Die Realität sieht leider ganz anders aus. Kalte Technik, Bitfriemelei, nervenaufreibende Probeläufe und immer wieder das Blättern im Handbuch gehören zum täglichen Brot der Assemblerprogrammierer.

Der Assembler

Zur Arbeitserleichterung wurde aber schon etwas getan. So verstehen Assemblerprogramme sogenannte Mnemonics. Dies sind kurze Wörter, bzw. Folgen von Buchstaben, die einen Maschinenbefehl beschreiben. Sie wurden als eine Art Orientierungshilfe entwickelt, da sich der Mensch Wörter, auch wenn sie manchmal nur aus scheinbar sinnlosen Buchstabenkombinationen bestehen, leichter merken kann als endlose Folgen von Nullen und Einsen.

Der Befehl

```
ADD #10,D0
```

wird von keinem Prozessor verstanden, erst ein Assembler setzt diesen Befehl in eine binäre Darstellung um. Es ist leicht einzusehen, daß sich ein Programmierer den Befehl ADD, er dient zur ADDition zweier Zahlen, leichter merken kann, als die äquivalente Darstellung in binärer Form. Diese lautet für den obigen Befehl:

1101 0001 1111 1000.

Diese binären Signale 'versteht' nun der Prozessor und addiert in Bruchteilen von einer Sekunde zu dem Datenregister D0 den Wert 10. Glücklicherweise muß sich kein Programmierer mehr um die binäre Darstellung von Maschinenbefehle kümmern, in vergangenen Zeiten war dies allerdings noch üblich.

Bleiben wir doch bei dem oben angesprochenen Beispiel. Hier war die Rede von einem 'Datenregister D0'. Daten- und Adreßregister sind wesentliche Bestandteile des im Atari ST eingesetzten MC68000 Prozessors. Diese Register sind eine Art RAM, also ein frei beschreibbarer Speicher innerhalb der CPU. Auf diese Register kann ähnlich wie auf eine Variable in Pascal oder Basic zugegriffen werden. Es sind mathematische Operationen wie Addieren, Subtrahieren, Dividieren und Multiplizieren möglich, sowie Vergleiche zum Testen zweier Wertigkeiten. Der Vorteil dieser Datenregister ist, daß ein Zugriff ungleich schneller geschieht als eine vergleichbare Operation im Hauptspeicher des Rechners.

Doch nicht nur mathematische Operationen sind hier von Bedeutung. Zusätzlich zu den 8 Datenregistern D0 bis D7 gibt es die 8 Adreßregister A0 bis A7. Mit ihnen können zwar auch die mathematischen Grundoperationen durchgeführt werden, jedoch ist ihr Haupteinsatzgebiet die Berechnung von Speicheradressen. Der Assemblerprogrammierer muß sich vor dem Schreiben seines Programmes im

klaren sein, wo er die zum Programmablauf zusätzlich benötigten Variablen und Parameter ablegen will. Um auf diese Werte später zugreifen zu können, gibt es insgesamt 14 verschiedene Möglichkeiten, Adressierungsarten genannt. Es gibt also 14 Varianten um Adreßregister zur Berechnung der Zieladresse einzusetzen.

Im weiter unten folgenden Schaubild sind die im MC68000 vorhandenen Register aufgeführt. Hier wird deutlich, daß die Daten- und Adreßregister eine Länge von 32-Bit haben, dies entspricht einem Langwort. Erfolgt ein Datenzugriff aber nur auf die unteren 16-Bit, handelt es sich um einen Wortzugriff. Wird auf die unteren 8-Bit zugegriffen, handelt es sich um einen Bytezugriff.

Die Unterscheidung der Operandenlänge ist wichtig für die Wahl der Adressierungsarten.

Zur besseren Übersicht nochmals eine Begriffserklärung der unterschiedlichen Datengrößen:

8-Bit = 1 Byte

Wertebereich: dezimal: 0-255
hex: 0-$FF

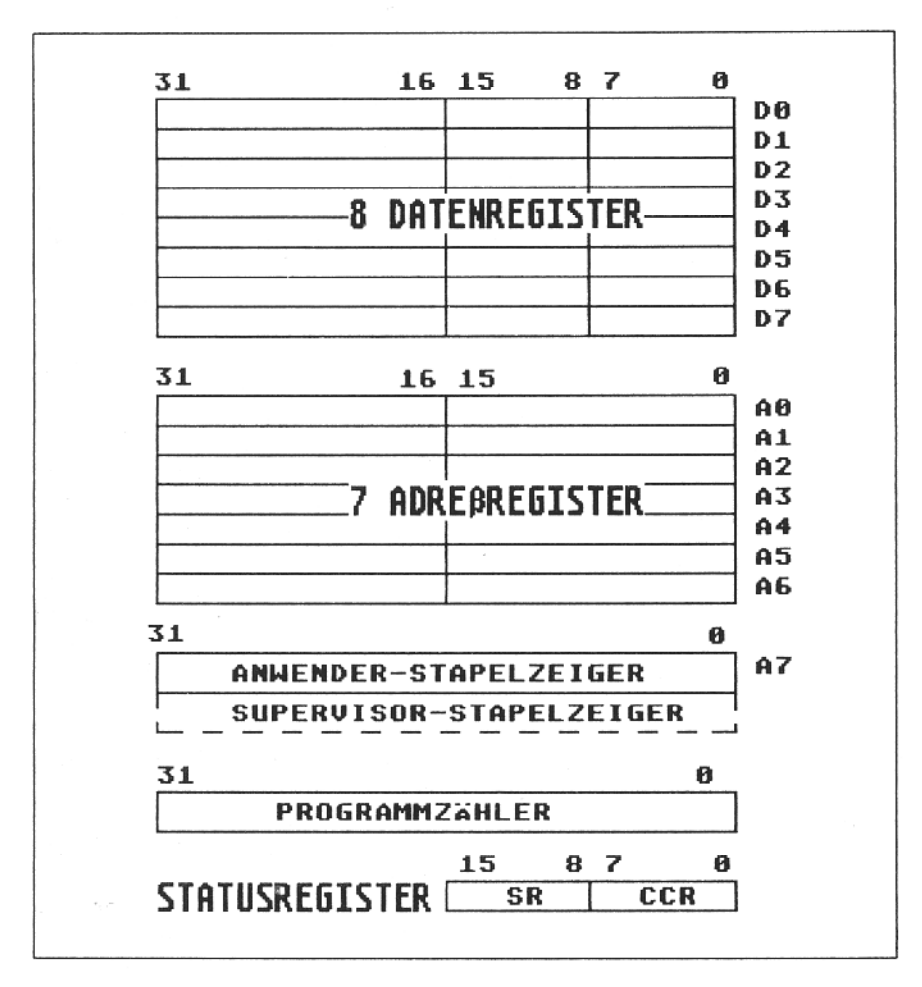
16-Bit = 2 Byte = 1 Wort
(engl. word)

Wertebereich: dezimal: 0-65536
hex: 0-$FFFF

32-Bit = 4 Byte = 1 Langwort
(engl. longword)

Zahlensysteme

Bei der Assemblerprogrammierung kommen in der Regel drei Zahlensysteme zur Anwendung: Das Dualsystem, daß Hexadezimalsystem und das dezimale Zahlensystem.



Dualsystem

Im Dualsystem werden die zwei Ziffern 0 und 1 verwendet, welche auch die Zustände darstellen, die ein Computer überhaupt nur unterscheiden kann. 0 steht für 'Aus' und 1 für 'An'. Die Kodierung einer 8-Bit Zahl erfolgt nach folgendem Schema:

Wertigkeit:							
128	64	32	16	8	4	2	1
Binär:							
0	0	0	1	0	1	0	0

Die Bitkombination 0001 0100 entspricht damit dem Wert 20.

Dezimalsystem:

Dies ist unser normales Zahlensystem mit den bekannten Ziffern 0 - 9.

Hexadezimalsystem:

Bei diesem System liegt die Basis 16 zugrunde, d.h. es müssen 16 verschiedene Ziffern existieren: 0-9 und A-F. Zur Kennzeichnung einer hexadezimalen Zahl wird dieser immer ein '$' vorangestellt.

dez: 9 10 11 12 13 14 15
hex: 9 A B C D E F

dez 10 = hex $A = bin 0000 01010

Der Stapelzeiger

In dem Registerschaubild tauchen aber noch weitere Namen auf. Der Programmzähler, auch ein 32-Bit Register, zeigt immer auf den nächsten auszuführenden Befehl. Im Statusregister wird der jeweilige Zustand der CPU anhand einiger Bits gekennzeichnet, die vom Anwender jederzeit abgefragt werden können. Interessant ist die Tatsache, daß das Adreßregister A7 offenbar eine doppelte Funktion hat. Dies hängt mit der Hardware der 68000er CPU zusammen. Es gibt zwei Betriebsarten, in denen die CPU ihren Dienst verrichten

kann. In der 'normalen' Betriebsart darf der Anwender nicht uneingeschränkt Betriebssystemadressen und Hardwarebausteine ansprechen. Nur wenn der Prozessor im sogenannten 'Supervisor-Mode' läuft, hat der Anwender die volle Kontrolle über den gesamten Speicherbereich. Je nach Betriebsart zeigt A7 auf einen anderen Speicherbereich. Dieser Speicherbereich soll nun näher beschrieben werden.

Im Schaubild ist A7 als Stapelzeiger (engl. Stackpointer) beschrieben. Der Programmierer darf das Register A7 nicht für seine Zwecke benutzen, da es schon eine feste Verwendung hat. In ihm steht die jeweils aktuelle Anfangsadresse einer Liste von Parametern, die vom Betriebssystem oder anderen Routinen benötigt werden. Rücksprungadressen von Unterprogrammen, Parameter für GEMDOS, BIOS oder XBIOS Routinen und auch Registerinhalte können hier temporär gespeichert werden, ohne daß sie von anderen Programmteilen überschrieben werden.

Bei jedem Neueintrag eines Wertes auf dem Stapel wird Adreßregister A7 um Eins vermindert. Der Stapel wächst nach unten, man spricht von einem Kellerstapel mit LIFO (Last In - First Out) Struktur. Um die langsam einsetzende Verwirrung zu vervollständigen sei bemerkt, daß die jeweilige Verminderung des Registers A7 von Eins nicht gleich Eins ist. Um welchen Wert das Register A7 bei Neueintrag eines Parameters auf den Stapel vermindert wird hängt entscheidend davon ab, welche Länge der verwendete Parameter besitzt.

Wird ein 16-Bit Wert (ein Wort) übergeben, so erfolgt eine Verminderung des Stapelzeigers um 2. Da der Speicher byteweise aufgebaut ist und ein 16-Bit Wert aus 2 Bytes besteht, wird dementsprechend der Zeiger um den Wert 2 korrigiert. Bei Übergabe eines Langwortes wird der Stapelzeiger um den Wert 4 vermindert. Diese Veränderung des Stapelzeigers geschieht automatisch durch die Verwendung der entsprechenden Adressierungsart.

Für Neueintragungen auf den Stapel wird die Adressierungsart 'Register indirekt mit Predekrement' verwendet, welche durch die folgende Schreibweise dargestellt wird:

-(An)

Für n können hier die Werte 0 bis 7, entsprechend der Adreßregister A0 bis A7 eingesetzt werden. Bezogen auf den Stapelzeiger muß die korrekte Verwendung natürlich

-(A7)

lauten, um die Verwendung des Adreßregisters A7 zu kennzeichnen.

Was aber bedeutet nun 'Register indirekt mit Predekrement' ? Der Begriff 'Register indirekt' deutet an, daß sich der zu verwendende Operand in der Speicherstelle befindet, auf die das Register A7 zeigt. Steht in A7 z.B. der Wert 5000, so beziehen sich sämtliche Operationen auf den Wert der sich in dieser Speicherstelle befindet.

Bei Verwendung der Variante 'Predekrement', wird VOR dem Datenzugriff der Inhalt von A7 vermindert.

Da ein genaues Verständnis der Arbeitsweise eines Stapels für die weitere Programmentwicklung unumgänglich ist, möchte ich hier ein kleines Beispiel einschieben um zu verdeutlichen, wie diese Adressierungsart in Verbindung mit dem Stapelzeiger verwendet wird. Hierzu wird noch ein weiterer Befehl nötig:

MOVE von,nach

Wer der englischen Sprache mächtig ist weiß, daß 'to move' bewegen oder verschieben bedeutet. Nun, da es bei einem Computer außer Daten nicht viel zum verrücken gibt, dürfte die Wirkungsweise diese Befehls schon auf der Hand liegen. Es werden Daten von einer Quelle zu einem bestimmten Ziel transportiert. Häufig dient dieser Befehl auch zum Laden eines Registers mit einem konstanten Wert. Mit

MOVE #1988,D0

wird das Datenregister D0 mit dem Wert 1988 geladen.

Die Quelldaten können aus einem direkt anzugebenden Wert bestehen, aus dem Inhalt einer Speicherstelle, oder aus dem Inhalt eines der vielen Register. Das Ziel wird in unserem unten aufgeführten Beispiel durch das Register A7, also den Stapelzeiger, bestimmt.

Die Übergabe eines Wertes auf den Stapel könnte demnach folgendermaßen aussehen:

MOVE #3000,-(A7)

Durch die Schreibweise #3000 wird verdeutlicht, daß der Wert 3000 direkt verwendet werden soll und nicht der Inhalt einer Speicherstelle.

Nun haben wir aber noch unsere Unterscheidung zwischen Wort und Langwort - woher weiß der Prozessor, ob er den Stapelzeiger nun um den Wert 2 oder um 4 vermindern soll? Wird der MOVE Befehl ohne weitere Parameter verwendet, so wird eine 16-Bit Verwendung angenommen. Die korrekte Schreibweise würde dann lauten:

MOVE.W #3000,-(A7)

Das .W steht für engl. word, und teilt dem Prozessor mit, daß hier ein 16-Bit Wert verwendet wird. Soll nun aber ein 32-Bit Wert übergeben werden, so muß dies auf jeden Fall im Befehl mit angegeben werden:

MOVE.L #3000,-(A7)

Schauen wir uns doch die Registerinhalte bei Ausführung der obigen Befehle genauer an. Es wird angenommen, daß das

angenommen, daß das Register A7 den Wert 5000 enthält, der Stapelzeiger A7 zeigt damit auf die Adresse 5000.

Der entsprechende Speicherbereich würde hierzu folgendermaßen aussehen (der Stapelzeiger ist durch einen Pfeil gekennzeichnet):

Adresse	Inhalt	
4996	0000	
4998	0000	
5000	7070	<--
5002	2500	
...		

Vor Ausführung des Befehls zeigt der Stapelzeiger auf die Adresse 5000, in der schon der Wert 7070 steht. Bei Ausführung des Befehls

MOVE.W #3000,-(A7)

wird zunächst der Inhalt des Registers A7 um den Wert 2 vermindert, da es sich bei 3000 um einen 16-Bit Wert handeln soll.

5000 - 2 = 4998

A7 enthält nun den Wert 4998 und verweist den Befehl MOVE an diese Speicherstelle.

4996	0000	
4998	0000	<--
5000	7070	
5002	2500	
...		

Dieser transportiert den im Befehl durch die Schreibweise #3000 angegebenen Wert an eben diese Speicherstelle, so daß sich folgendes Speicherbild ergibt:

4996	0000	
4998	3000	<--
5000	7070	
5002	2500	
...		

Natürlich ist das Übergeben eines Wertes auf den Stapel nur die halbe Miete. Wird der Wert nicht mehr benötigt, muß der Stapelzeiger entsprechend nach oben korrigiert werden. Ohne diese notwendige Korrektur würde der Zeiger quer durch den Speicher irren und wichtige Systemvariablen zerstören - die Folge wäre ein Absturz des Systems und die obligatorischen Bomben.

Die Korrektur des Stapelzeiges kann auf zwei Arten erfolgen. Werden die Werte in den Speicherstellen nicht mehr benötigt, kann der Zeiger einfach mit dem Befehl

ADD.L #Wert,A7

um die nötige Anzahl von Speicherstellen geändert werden. Hierbei ist zu beachten, daß die Addition immer als 32-Bit Operation erfolgen muß, da der Stapelzeiger auch auf Speicherplätze zeigen kann, die sich nicht mehr mit 16-Bit Zahlen darstellen lassen.

Die Speicherstelle $8FFFF wäre als 16-Bit Wert gesehen nur $FFFF. Bei einer Wort-Addition mit dem Wert 2, ergäbe sich ein 16-Bit Wert von $0001. Die anschließende Operation würde sich auf die Speicherstelle 1 beziehen. Bei einer exakten 32-Bit Operation wird der Überlauf entsprechend berücksichtig, und es ergibt sich der Wert $90001.

Für jeden auf den Stapel übergebenen 16-Bit Wert wird 2 addiert und für jeden 32-Bit Wert 4. Durch Verwendung des Befehls ADD.L #2,A7 ergibt sich folgende Veränderung des Spapelzeigers:

4996	0000	
4998	3000	<--
5000	7070	
5002	2500	
...		

Die Abarbeitung von ADD.L #2,A7 ergibt:

4996	0000	
4998	3000	
5000	7070	<--
5002	2500	
...		

Die Werte der Speicherstellen 4998 und 5000 werden, da sie nicht mehr benötigt werden, einfach übersprungen. Ein Löschen ist hier nicht nötig.

Die zweite Art um den Stapelzeiger wieder zu korrigieren, geschieht durch eine weitere Adressierungsart: Adreßregister indirekt mit Postinkrement, dargestellt durch die Schreibweise

MOVE.W (A7)+,D0

Zur Berechnung der gewünschten Adresse wird auch hier wieder der Inhalt des angegebenen Adreßregisters, hier der Stapelzeiger, verwendet. Dies wird beschrieben durch die Formulierung 'Adreßregister indirekt'. Postinkrement ist das genaue Gegenteil von Predekrement. Wird bei letzterem das Adreßregister VOR der Adreßberechnung vermindert, so wird durch ein Postinkrement das Adreßregister NACH Ausführung des Befehls erhöht. Die Veränderung des Inhaltes von A7 geschieht ebenfalls automatisch und ist abhängig von der verwendeten Operandengröße.

Der Vorteil bei dieser Methode ist, daß die zuvor auf dem Stapel abgelegten Werte weiterverarbeitet werden können. So können Registerinhalte kurzfristig zwischengespeichert werden und bequem wieder eingeladen werden.

Im Beispiel wird die Arbeitsweise verdeutlicht:

4996	0000	
4998	3000	<--
5000	7070	
5002	2500	
...		

Durch den Befehl MOVE.W (A7)+,D0 ergibt sich:

```
4996    0000
4998    3000
5000    7070  <--
5002    2500
...
```

Der Inhalt aus Adresse 4998 wurde in das Datenregister D0 kopiert, so daß der Wert 3000 nun zu weiteren Berechnungen in einem Register zur Verfügung steht.

2. Betriebs- systemaufrufe

Das Betriebssystem des Atari ST stellt dem Assemblerprogrammierer eine umfangreiche Sammlung von Unterprogrammen zur Verfügung. Mit ihnen lassen sich praktisch alle grundlegenen Funktionen des Atari ST programmieren.

Die unterste Ebene bilden die BIOS und XBIOS Funtionen, eine Stufe darüber die sind die GEMDOS Funktionen implementiert. Dies bedeutet, daß BIOS und XBIOS auf keine anderen Funktionen mehr zurückgreifen und die Hardwarebausteine direkt ansprechen. Eine genaue Erklärung der wichtigsten Funktionen erfolgt im Rahmen diese Artikels mit entsprechenden Beispielen.

Zunächst gilt es aber die Frage zu klären, wie ein solcher Aufruf einer Betriebssystemroutine geschieht. Vier Schritte sind hierbei nötig:

1. Übergabe der Parameter auf den Stapel
2. Übergabe der Funktionsnummer
3. Aufrufen der Funktion mit dem Befehl TRAP #n
4. Restaurieren des Stapelzeigers

Die Punkte 1,2 und 4 werden durch die schon oben dargestellte Stapeltechnik ausgeführt, so daß der Punkt 3 noch zu klären wäre. Bei Erreichen des Befehls TRAP geht der Prozessor in einen Ausnahmezustand über, d.h. der normale Programmablauf wird unterbrochen und es wird ein Unterprogramm abgearbeitet. Welches Unterprogramm abgearbeitet wird, hängt von dem zusätzlichen Parameter des Befehls ab.

Insgesamt kann der MC68000 16 TRAPs verwalten, beim Atari ST werden jedoch nur die folgenden 4 genutzt:

TRAP #1 - GEMDOS
TRAP #2 - AES & VDI (GEM)
TRAP #13 - BIOS
TRAP #14 - XBIOS

Über einen der jeweiligen Trapnummer zugehörigen Vektor wird ein Trap-Dispatcher aufgerufen. Dieser erkennt die in Schritt 2 übergebene Funktionsnummer und leitet die Programmkontrolle weiter. Nach erfolgter Abarbeitung der jeweiligen Funktion kehrt der Programmablauf wieder in das Anwenderprogramm zurück und übergibt eventuelle Parameter oder Fehlermeldung in den Registern D0 oder A0 zurück.

Im Zusammenhang der Registerbenutzung ist hierbei zu beachten, daß die Inhalte der Datenregister D0 - D2 und der Adreßregister A0 - A2 verändert werden und bei Bedarf auf dem Stapel zwischengespeichert werden sollten.

Drei Beispiele

Im folgenden Beispiel werden drei GEMDOS Routinen vorgestellt und in ihrer Arbeitsweise erklärt. Es handelt sich zunächst um einen GEMDOS Aufruf, mit dessen Hilfe eine Zeichenkette auf dem Bildschirm ausgegeben wird. Diese Zeichenkette steht als Konstante am Ende des Programmes. Damit der Assembler bei der Programmübersetzung diese nicht als Befehl interpretiert, haben Assembler spezielle Befehl um Zeichenketten oder ähnliche Konstanten als solche zu deklarieren. Die gebräuchlichste Form ist der Befehl:

DC.B (auch .DC.B)

Die genaue Schreibweise entnehmen sie bitte dem jeweiligen Assemblerhandbuch.

Wichtig ist, daß am Ende der Zeichenkette ein Null-Byte steht, damit das Ende eindeutig gekennzeichnet ist. Als erstes wird nun in unserem Beispielprogramm die Startadresse der Zeichenkette auf den Stapel übergeben. Damit wir die Startadresse der Zeichenkette nicht von Hand berechnen müssen, definieren wir ein sogenanntes Label. Der Assembler berechnet dann während der Programmübersetzung automatisch den einzusetzenden Wert. Auch hier kann die Deklaration eines Labels von Assembler zu Assembler abweichen.

Bitte beachten sie auch, daß hier ein 32-Bit Wert übergeben werden muß, um auch Adressen größer als $FFFF (= max. 16-Bit Wert) ansprechen zu können.

Als zweiter Parameter wird die Funktionsnummer der gewünschten Routine auf dem Stapel abgelegt. Für die Funktion 'PRINT LINE' ist der Wert 9 als 16-Bit Wert zu übergeben. Nun erfolgt durch den Befehl TRAP #1 eine Unterbrechung des normalen Programmablaufes und eine Verzweigung in den GEMDOS Bereich des TOS. Hier wird die zuvor auf dem Stapel abgelegte Adresse der auszugebenden Zeichenkette weiterverarbeitet und der Text 'Hello world' erscheint auf dem Bildschirm.

Nach erfolgreicher Ausführung kehrt der Programmablauf zurück und mit dem Befehl

ADD.L #6,A7

wird der Stack entsprechend der zuvor abgelegten Werte korrigiert.

Die Zahl 6 ergibt sich aus der 32-Bit Adresse (= 4 Bytes) und dem 16-Bit Funktionswert (= 2 Bytes).

Bei diesem Befehl darf keine Klammer um A7 gesetzt werden, da der Wert direkt zum Adreßregister A7 addiert werden soll. Wäre eine Klammer gesetzt, so hätten wir eine indirekte Adressierung und es würde der Inhalt einer Speicheradresse auf dem Stapel um 6 erhöht werden.

Die zweite Funktion ist 'Conin' und liest ein Zeichen von der Tastatur ein. Als Parameter wird hier nur die Nummer der Funktion (= 1) übergeben. Die Routine wartet nun, bis der Anwender eine Taste gedrückt hat und kehrt erst dann zum normalen Programmablauf zurück.

Den ASCII Wert der gedrückten Taste bekommen wir in den untersten 8-Bit des Datenregister D0 zurück, für unser Beispiel ist es jedoch gleichgültig welche Taste gedrückt wurde, so daß wir den Wert nicht weiter beachten. Auch hier muß nach der Rückkehr mit dem entsprechenden Befehl der Stapelzeiger wieder korrigiert werden.

Der dritte Funktionsaufruf schließlich dient zum verlassen des Programmes und sorgt für eine Rückkehr zum Destop. Eine Korrektur des Stapelzeigers ist hier nicht nötig und gar nicht möglich, da bei erreichen des Befehls TRAP #1 unser Programm bereits beendet ist. Die Funktionsnummer für die Routine 'Pterm' lautet 0 und wird als 16-Bit Wert auf dem Stapel übergeben.

Der Sourcecode des Beispielprogrammes ist im folgenden aufgeführt. kuw

```
MOVE.L #Adresse,-(A7)
MOVE.W #9,-(A7)
TRAP   #1
ADD.L  #6,A7

MOVE.W #1,-(A7)
TRAP   #1
ADD.L  #2,A7

MOVE.W #0,-(A7)
TRAP   #1

Adresse:
 DC.B 'Hello world',0
```

GEMDOS Aufruf Nr. 0 - PTERM

FUNKTIONSNUMMER: 0
PARAMETER: keine
RÜCKGABE: keine

Beendet das laufende Programm und kehrt in das Programm zurück, von dem der Programmstart erfolgte. Dies wird in den meisten Fällen das Desktop sein Da ein Programmstart aber auch von einer anderen Anwendung aus erfolgen kann, z.B. einer Shell, ist eine Ausführung dieser Routine nicht immer mit der Rückkehr zum Desktop verbunden.

Beispiel:

```
MOVE.W  #0,-(A7)     * Funktionsnummer
TRAP    #1           * GEMDOS aufrufen
* Hier kommen wir niemals hin....
```

GEMDOS Aufruf Nr. 1 - CONIN

FUNKTIONSNUMMER: 1
PARAMETER: keine
RÜCKGABE: ASCII-Wert und SCAN-Code in Datenregister D0 zurück

Diese Funktion liest ein Zeichen von der Tastatur ein, dabei wird solange gewartet, bis eine Taste gedrückt wurde. Der Rückgabewert ist ein 32-Bit Wert und wird im Register D0 übergeben. In den Bits 0-7 steht der ASCII-Wert der gedrückten Taste. Da nicht alle auf der Tastatur vorhandenen Tasten ein ASCII-Wert besitzen, wird zusätzlich in den Bits 16-23 der SCAN-Code der jeweiligen Taste übergeben.

Beispiel:

```
MOVE.W  #1,-(A7)     * Funktionsnummer
TRAP    #1           * GEMDOS aufrufen
ADD.L   #2,A7        * Stapel korrigieren
```

GEMDOS Aufruf Nr. 9 - PRINT LINE

FUNKTIONSNUMMER: 9
PARAMETER: Adresse einer mit Null abgeschlossenen Zeichenkette
RÜCKGABE: Anzahl der ausgegebenen Zeichen im Datenregister D0

Ausgabe einer Zeichenkette ab der aktuellen Cursorposition auf dem Bildschirm. Diese muß mit einem Null-Byte abgeschlossen sein.

Beispiel:

```
MOVE.L  #Adresse,-(A7)  * 32-Bit Adresse
MOVE.W  #9              * Funktionsnummer
TRAP    #1              * GEMDOS
ADD.L   #6,A7           * Stapel korrigieren
```

STARKE SOFTWARE



DAS PROGRAMM FÜR SYMBOLISCHE ALGEBRA UND ANALYSIS

LEISTUNGSMERKMALE:

- rechnet ohne Rundungsfehler
- verarbeitet symbolische Ausdrücke wie 2x ∂y ∂z
- löst Gleichungen nach beliebigen Variablen auf
- beherrscht Grenzwerte, Differential- und Integralrechnung
- ideal für Matritzenrechnung
- unglaublich schnell, da **vollständig** in-Assembler geschrieben
- nicht nur ein Mathematik-Programm, sondern gleichzeitig eine komplette KI-Sprache, die leicht erlernbar ist
- die Kombination von Mathematik-System und Programmiersprache ermöglicht auch Anfängern komplexe Mathe-Programme mühelos zu schreiben
- einfache und komfortable Bedienung
- für Schüler, die sich Fehlrechnungen ersparen wollen
- für Lehrer, die mit ST-MATH Klausuren mühelos und schnell korrigieren wollen
- für Studenten, die lange Umformungen und Rechnungen zeitsparend durchführen wollen
- für Ingenieure und sonst. Anwender, die oft komplexe nichtnumerische Probleme lösen müssen
- für jeden, der eine preiswerte, aber dennoch vollwertige Sprache für künstliche Intelligenz sucht, die leicht zu erlernen ist
- **Hardware:** ATARI ST mit 512 KB und ROM-TOS oder 1 MB (dann auch RAM-TOS möglich), eins. Disk.-Laufwerk
- arbeitet mit Farb- und SW-Monitoren
- Lieferung mit umfangreichem deutschen Handbuch, das die Mathematikfunktion detailliert erläutert und zugleich eine ausführliche Einführung in die Programmierung von ST-MATH enthält.

EIN SPITZENPROGRAMM **DM 98.–**



KURVENDISKUSSIONS- U. FUNKTIONSPLOTTER-PROGR.

LEISTUNGSMERKMALE

- **ST-PLOT** ist ein Programm, das mehr kann, als einfach nur Funktionen zeichnen. Es bestimmt die richtigen Ableitungen und damit werden Null-, Extrem- und Wendestellen berechnet, es stellt den Definitionsbereich und die Periode fest und vieles andere mehr.
- **ST-PLOT** zeichnet die Schaubilder von bis zu 3 Funktionen und deren erste und zweite Ableitung gleichzeitig auf einen Bildschirm. Danach können Ausschnitte der Funktionen vergrößert und verkleinert werden.
- **ST-PLOT** ist komplett GEM- und mausgesteuert. Die Bedienung ist somit sehr einfach. Sie brauchen nur die Funktion einzugeben, alle anderen Berechnungen erledigt für Sie das Programm
- **ST-PLOT** läuft sowohl in mittlerer wie auch in hoher Auflösung
- **ST-PLot** ist nicht nur für Professoren, Studenten und Schüler interessant. Es kann überall dort angewendet werden, wo man sich mit Funktionen beschäftigt.

ZUR EINGABE

- Funktionslänge max. 210 Zeichen
- Verwendung sämtlicher Funktionen eines wissenschaftl. Rechners
- Trigonometrische und hyperbolische Funktion und deren Umkehrfunktion; damit können auch komplexere Funktionen wie z.B. Fourierreihen und Taylorentwicklungen berechnet werden, weiterhin Logarithmen, Exponentialfunktionen, Wurzel, Betrag etc.
- Definition von 10 verschiedenen Zahlenwerten als Konstanten (Bei Funktionseingabe wird dann nur die Konstante eingegeben)
- e und Pi verwendbar

keine Eingabe des Multiplikationszeichens. (Nachträgliche Einsetzung)

- Abspeichern der Funktionen in DEGAS und SCREEN-Format
- Eingabe nicht nur von Funktionen, sondern auch von BELIEBIGEN KURVEN (Relationen möglich.

ZUR ZEICHNUNG

- Automatische Zeichnung und Beschriftung der Achsen und deren Skalierung.
- **Fehlerroutinen, damit keine unnötigen Asymptoten gezeichnet werden, sondern nur die Funktion**
- **Schrittweite ist in Abhängigkeit der Steigung der Funktion**
- Wahlweise Rasterlegung über den Bildschirm möglich, wie die Koordinaten, unter dem Mauszeiger angezeigt werden.
- Beliebig viele Funktionen nacheinander in ein Bild einzeichenbar. Damit Vergleich unterschiedlicher Funktionen und Ablesung der Schnittpunkte möglich.
- Anschluß eines PLOTTERS
- Bessere Unterscheidung der Funktionen und deren Ableitung durch unterschiedliche Liniendarstellung

ZUR BERECHNUNG DER FUNKTIONSWERTE:

- Hohe Rechengeschwindigkeit durch sehr kleine Routinen, die die Funktionswerte berechnen.
- Hohe Rechengenauigkeit durch numerische Berechnung der Ableitungen

DM 69.–

HARD COPY ST PLOT ▶



LOGIKSIMULATOR FÜR DEN ATARI ST

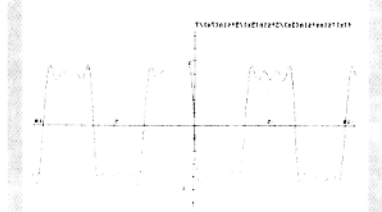
Ein Programm zum Erstellen, Testen und Analysieren von Logikschaltungen für Ausbildung und Hobby-Elektronik

LEISTUNGSMERKMALE

- Komfortable GEM-Umgebung
- Bauteile lassen sich per Maus plazieren und verdrahten
- Umfassendes Bauteile-Set (Grundgatter, Ein-/Ausgabe-Bausteine, Flip-Flops, etc.
- Definition zusätzlicher Bauteile durch Makrotechnik
- Makros können in Libraries gespeichert werden
- Interaktive Simulation mit Darstellung der Leitungszustände (d.h. Betätigung von 'Schaltern' mit der Maus und sofortige Reaktion der Schaltung)
- Erzeugen von Impulsdiagrammen
- Hardcopy-Funktion
- Hardwarevoraussetzung:
- ST/Monochrom-Monitor
- gängige Druckertreiber vorhanden









NEU

SIMULATION VON ANALOGSCHALTUNGEN

Das Simulations-Programm zum Analysieren, Testen und Entwickeln von analogen Elektronikschaltungen (Kettenschaltungen) für Hobby, Ausbildung und Studium

LEISTUNGSMERKMALE

- Komfortable Maus-Steuerung
- Grafischer Schaltungsaufbau
- Einfache Eingabe und Änderung von Bauteilwerten
- Max. 65 Bauteile pro Schaltung (z.B. Stromquellen, Übertrager, Schwingkreise, offene und kurzgeschlossene Stichleitungen, Übertragungsleitungen usw.)
- „Wobbel-Generator" von 1 Hz bis 2 GHz
- Max. 0,1 Hz bzw. 1 Hz Auflösung
- Wahlmöglichkeit zwischen sehr schneller oder sehr genauer Berechnung
- Grafische Ausgabe von Spannungs- und Stromverhältnissen, von Phasenverläufen und von Eingangsimpedanzen
- logarithmische und lineare Koordinatenachsen
- Verstärkung und Offset einstellbar
- Einfaches Testen der Schaltung im Rückwärtsbetrieb
- Digitale Anzeige von Funktionswerten bei diskreten Frequenzen
- Hardcopy-Funktion
- **Hardwarevoraussetzung:** ST mit Monochrom-Monitor
- ausführliches deutsches Handbuch

DM 98.–

ST SALIX DIALOGEXPERTE

Wer schon einmal eine Anwendung unter GEM geschrieben hat – mit Drop-down-Menüs, Dialogboxen und Alerts, wird es wissen:
So einfach sich die einzelnen Objekte mit dem Resource Construction Set Editor erzeugen lassen, so umständlich ist ihre Einbettung in das Anwendungsprogramm.
Es geht auch anders: SALIX hat jetzt einen DIALOG-Experten entwickelt, der die RSC- und die DEF-Dateien selbständig untersucht. Dann geht's erst richtig los: **„Programming by doing"** heißt die Device. Zunächst werden die Fenster der Anwendung definiert – mit der Maus. Dann wird die Bedeutung der Menüpunkte festgelegt, einfach indem man sie anklickt und das aufzurufende Prädikat eingibt.
Mehrere Menüleisten sind zulässig. Die einzelnen Menüpunkte lassen sich in logische Gruppen zusammenfassen, das Check-Symbol wird kontrolliert, und und und ...
Die Knöpfe der Dialogboxen und Alerts lassen sich symbolisch ansprechen, eine Reihe von Prädikaten ermöglicht den einfachen Zugriff auf die GEM-Objekte von der Anwendung heraus aufgebaut werden.
Sie haben die Wahl: sich weiterhin damit abzuquälen, Menüpunkte auszuzählen, oder sich lieber auf das Design und Logik Ihrer Anwendung zu konzentrieren.
Der Dialogexperte läuft unter SALIX PROLOG 2.6.

DM 79.–

Alle Preise sind unverbindlich empfohlene Verkaufspreise

Erhältlich bei Ihrem Computer-Händler

Programme mit hohem Niveau

Heim Verlag

GfA-BASIC VERSION 3.0

Ein Traum ist endlich Realität
Programmieren kinderleicht gemacht

Auf der CeBit 88 wurde das GfA-Basic 3.0 dem Publikum vorgestellt. Nun wartete man gespannt auf die endgültige Version, um sich aus seinem altem GfA-BASIC ein neues machen zu lassen.

Update

Mittlerweile ist sogar schon die Version 3.01 auf dem Markt. Hier ist das Problem der Updates, das viele Programme auf dem Atari ST betrifft: Die ersten Versionen beinhalten noch kleine Fehler, die erst nach und nach behoben werden. Aber dies Art von Produktpolitik soll auch der einzige kleine Schönheitsfehler sein.

Nun, was hat die Version 3.0 dem Programmierer neues zu bieten ?

Der Editor

Hier hat sich die Menüleiste ein wenig verändert. Links neben der etwas zusammengeschobenen Leiste steht das Atari-Zeichen. Nach dem Anklicken dieses Zeichens erscheint ein Pull-Down-Menü. Hier hat man Zugriff auf Accessories, kann Programme sowohl laden als auch speichern und DEFLIST einstellen. Außerdem enthält es die Funktion 'Neue Namen'.

Neue Namen

Hiermit werden beim Schreiben von Programmen die eingegebenen Variablennamen dahingehend überprüft, ob es sich um bisher noch nicht benutzte Variablennamen handelt. Bei einem neuen Namen oder einem Schreibfehler erscheint eine Warnmeldung. So läßt sich diese Art von Fehlern direkt bei der Erstellung von Programmen vermeiden.

Bei der Fehlersuche ist dieser Fehler nur sehr schwer zu finden, da nicht jeder bei der Fehlersuche auf die genaue Schreibweise aller Variablen achtet.

Bisher konnte man solche Fehler nur mit dem Public-Domain-Programm XREF finden (siehe unsere PD Nr. 128). Es listet alle benutzten Variablen mit den jeweiligen Zeilennummern auf. Wird eine Variable nur ein einziges mal benutzt, kann irgend etwas nicht stimmen.

Zeit ist Geld

In der rechten Ecke der Hauptmenüleiste läuft nun eine Uhr mit. Diese kann durch Anklicken und Eingabe der neuen Zeit gestellt werden. Unter der Uhr befindet sich die Anzeige der Zeilennummer. Durch einfaches Anklicken der Zeilennummer springt der Cursor in die neu einzugebende Zeile.

Befehle über die Tastatur

Sämtliche Kommandos können auch mit der Tastatur gesteuert werden, wobei eine Funktion über verschiedene Tasten aufgerufen werden kann. Um im Sourcetext eine Seite nach oben zu blättern, stehen dem Anwender gleich 4 Tastenkombinationen zur Verfügung:

- Shift+'F7'
- Control+'R'
- Control+'Cursor oben'
- Ctrl+'9 auf dem Ziffernblock'

Hier kann sich jeder Programmierer seinen idealen Funktionsaufruf merken.

Generell kann der Ziffernblock auf Cursorsteuerung umgeschaltet werden (Numlock), so daß dieser dann vergleichbar mit denen der PC's ist.

Definition der Funktionstasten

Ebenfalls neu ist die Belegung der Funktionstasten durch Texte. Hier können die wichtigsten 20 Befehls- oder Variablennamen mit je maximal 31 Zeichen gespeichert und bei Bedarf mit Alternate abgerufen werden.

Das Setzen der Belegung erfolgt am besten durch ein kleines BASIC-Programm, welches die Texte enthält und einmal gestartet wird. Der Befehl zur Belegung heißt übrigens KEYDEF.

Das Bildschirmscrollen erfolgt bei der neuen Version merklich schneller. Zum Auffinden eines Programmstückes stehen jetzt sechs Marken zur Verfügung, weitere vier Marken sind fest definiert (Stelle der letzten Änderung, etc.).

Proceduren

Eine weitere wichtige Neuerung ist das Einklappen von Proceduren. Hiermit kann das zu bearbeitende Programm übersichtlicher gestaltet werden. Es

erscheint im Listing nur noch PROCEDURE <Procedurename>. Der zu der Procedure gehörende Programmtext wird also unterdrückt.

Wird eine Änderung an der Procedure gewünscht, so kann diese zur weiteren Bearbeitung aufgeklappt werden. Alle übrigen im Programm vorhandenen Proceduren bleiben aber *verschlossen*. Der Programmtext wird dadurch übersichtlicher und man findet schneller die gesuchten Programmteile.

Leichtes Spiel mit Fehlern

Auch bei den Hilfen zur Fehlersuche haben sich etliche Neuerungen ergeben. Mit

TRON <proc>

wird nach jedem abgearbeiteten Befehl in die Procedure <proc> verzweigt. Der Anwender hat hier die Möglichkeit, sich eine eigene, den jeweiligen Bedürfnissen entsprechende Routine zu schreiben. So kann sich der Anwender Variableninhalte anzeigen lassen oder sonstige Überprüfungen der Variablen vornehmen, um einen eventuellen Fehler schneller zu finden und zu beseitigen.

Um eine Art von Multitasking zu realisieren, kann eine regelmäßige Programmunterbrechung durch den Befehl

EVERY ticks GOSUB <proc>

erfolgen. Daraufhin wird in einem zeitlichen Abstand von ticks/200 Sekunden die Procedure <proc> abgearbeitet.

Diese Interruptprogrammierung kann auch zur Fehlersuche verwendet werden. Eine Art von Multitasking wäre zum Beispiel die regelmäßige Anzeige der Uhrzeit.

Ein einmaliges Bearbeiten der Procedure <proc> erfolgt durch folgenden Aufruf:

AFTER ticks GOSUB <proc>.

Der Befehl

DUMP

sorgt für eine Auflistung aller im Programm benutzten Variablennamen, Proceduren und Labels.

Zur weiteren Fehlerbehandlung stehen die bekannten Befehle

ON ERROR und
ON BREAK

zur Verfügung.

BASIC-Befehle

Im BASIC selbst hat sich natürlich viel getan. Jetzt stehen insgesamt sechs Variablentypen zur Verfügung. Dies sind neben Strings, Boolean, Integer und Float noch die beiden neuen Typen BYTE und WORD.

Dabei kann BYTE die Werte von 0 bis 255 annehmen - für den Variablentyp WORD reicht der darstellbare Zahlenbereich von -32768 bis 32767. Wie sich aus dem Namen der Variablen schon vermuten läßt, belegt der Variablentyp BYTE ein Byte - der Type WORD hingegen zwei Byte im Speicher.

Geändert hat sich der Speicherbedarf einer Float-Variablen von sechs auf acht Byte. Damit ist der Zahlenbereich auf

2.225073858507E-308 bis
3.595386269725E+308

gewachsen (13 Stellen Genauigkeit). Von allen Variablentypen lassen sich Felder erzeugen.

Sortierung

Felder könen nun auch sortiert werden, ohne das der Anwender die hierzu nötigen mathematischen Verfahren selbst entwickeln muß. Hierzu stellt das neue GfA-BASIC gleich zwei Verfahren zur Verfügung, die auf zwei unterschiedlichen Algorithmen aufgebaut sind.

Hier nun eine Sammlung aller Befehle und Funktionsnamen, die das GfA-BASIC 3.0 enthält:

!
$
'
()
*
+
-
.
/
<=
<>
<
=<
==>
==
=>
=
><
>=
>
?
@

ABS
ABSOLUTE
ACHAR
ACLIP
ACOS
ADD
ADDRIN
ADDROUT
AFTER
AFTER CONT
AFTER STOP
ALERT
ALINE
AND
APOLY
APPL_EXIT
APPL_FIND
APPL_INIT
APPL_READ
APPL_TPLAY
APPL_TRECORD
APPL_WRITE
ARECT
ARRAYFILL
ARRPTR
ASC
ASIN
ATEXT
ATN

BASEPAGE
BCHG
BCLR
BGET
BIN$
BIOS
BITBLT
BLOAD
BMOVE
BOUNDARY
BOX
BPUT
BSAVE
BSET
BTST
BYTE

C:
CALL
CARD
CASE
CFLOAT
CHAIN
CHAR
CHDIR
CHDRIVE
CHR$
CINT
CIRCLE
CLEAR
CLEARW
CLIP
CLOSE
CLOSEW
CLR
CLS
COLOR
CONT
CONTRL
COS
COSQ
CRSCOL
CRSLIN
CVD
CVF
CVI
CVL
CVS

DATA
DATE$
DEC
DEFAULT
DEFBIT
DEFBYTE
DEFDBL
DEFFILL
DEFFLT
DEFFN
DEFINT
DEFLINE
DEFLIST
DEFMARK
DEFMOUSE
DEFNUM
DEFSGN
DEFSTR
DEFTEXT
DEFWRD
DEG
DELAY
DELETE
DFREE
DIM
DIR
DIR$
DIV
DO
DO UNTIL
DO WHILE
DOUBLE
DOWNTO
DPEEK
DRAW
DUMP

EDIT
ELLIPSE
ELSE
ELSE IF
END
ENDDO
ENDFOR
ENDFUNC

QSORT

arbeitet nach dem Quicksort-Verfahren und

SSORT,

welches das Shellsort-Verfahren als Algorithmus benutzt. Dabei kann jeweils ein beliebiges Array sortiert werden. Zusätzlich können folgende Parameter angegeben werden:

Durch ein Plus- oder Minuszeichen kann die Reihenfolge der Sortierung bestimmt werden.

Eine Zahl n, so daß nur die ersten n Elemente des Arrays sortiert werden.

Ein Integer-Array, welches bei der Sortierung mitsortiert wird (wichtig für Adressenverwaltung etc.)

Ein weiteres Integer-Array mit mindestens 256 Elementen, welches als Sortierkriterium dient. Hier kann die Reihenfolge der Sortierung von Elementen selbst bestimmt werden, um z. B. die Sonderzeichen Ä, Ö, Ü und ß ins Alphabet richtig einzusortieren. Somit kommt Schäfer wirklich vor Schulz und nicht hinter Schwarz.

Man beachte, daß es sich immer noch um BASIC handelt !

BIT für BIT

Ein ganz besonderer Leckerbissen ist die Verwendung von Bitoperationen. Jetzt können wie in einem Maschinenprogramm einzelne Bits gesetzt, gelöscht, negiert und abgefragt werden. Weiterhin gibt es Schiebe- und Rotierbefehle, mit der die Datentypen Byte, Word und Long verändert werden können.

Ein beliebtes Beispiel ist in diesem Zusammenhang die Multiplikation mit 2, welche jeder Assemblerprogrammierer mit einer kurzen Verschiebeoperation erledigt. Basicprogrammierer mußten dies bisher auf konventionelle Art und Weise lösen:

X%=X%*2

Durch Verwendung des binären Schiebebefehls 'shift left', ergibt sich folgende Schreibweise:

X%=SHL(X%,1)

So wird der zeitaufwendige Multiplikationsbefehl durch einen wesentlich schnelleren Schiebebefehl ersetzt.

Die Vielfalt der neuen Befehle scheint keine Grenzen zu kennen. Die vielen Strukturbefehle sind sicher der Höhepunkt . Es stehen neben der bisher bekannten Vergleichsfunktion

IF

nun auch die von höheren Programmiersprachen bekannten Strukturen wie

ELSE IF,
SELECT,
CASE TO,
DEFAULT,
OTHERWISE,
ENDSELECT und
CONT

zur Verfügung.

Um es den Umsteigern anderer Programmiersprachen einfach zu machen, wurden auch neue Befehle wie

ENDFOR,
ENDREPEAT,
ENDWHILE und
ENDDO

eingeführt, die mit

NEXT,
UNTIL,
WEND und
LOOP

identisch sind. Ebenfalls identisch ist die Verwendung von

RETURN und

ENDPROC,
ENDSUB.

Aus einer simplen

DO LOOP - Schleife

kann nun

DO WHILE,
DO UNTIL,
LOOP WHILE und
LOOP UNTIL

gemacht werden. Hier muß man die Abbruchbedingung nicht mehr negieren, z. B. statt

```
REPEAT
  ...
UNTIL NOT (A>B OR C>D)
```

kann auch

```
DO
  ...
LOOP WHILE A>B OR C>D
```

geschrieben werden.

Proceduren

Endlich ist es möglich, im Procedureaufruf nicht nur eine Variable sondern auch einen Variablennamen zu übergeben, ähnlich wie es in PASCAL üblich ist. Bisher mußte man entweder den Pointer (die Adresse, in der der Wert der Variablen steht) übergeben, oder es mußte auf globale Variablen zurückgegriffen werden, falls die Procedure einen Wert zurückgibt.

Ebenfalls möglich sind nun mehrzeilige Funktionen, in denen ein berechneter Wert zurückgegeben wird. Die mehrzeilige Funktion kann an mehreren Stellen beendet werden.

Grafik

Auch in der Grafikprogrammierung kamen einige neue Befehle dazu. Dies ist z. B.

BOUNDARY,

um die Umrahmung bei einer Füllfläche ein- und auszuschalten. Hinter

DEFTEXT

und den bisher bekannten vier Parametern kann nun noch ein fünfter folgen, der den Zeichensatz angibt, der natürlich vorher geladen werden muß. Mit

CLIP

kann die Grafikausgabe auf ein festgelegtes Rechteck begrenzt werden. Hinter

DRAW

können nun Befehle folgen, die der Ansteuerung eines Plotters, bzw. den Grafikbefehlen der Programmiersprache LOGO ähneln. So können Bewegungen eines imaginären Stiftes absolut und relativ erfolgen. Durch einen Winkel wird die Richtung angegeben, in der gezeichnet werden soll.

Immer schneller

Schnelle Grafik erfolgt durch die LINE-A Routinen, die in GfA 3.0 implementiert wurden. Mit

PSET

lassen sich Punkte in einer bestimmten Farbe auf den Bildschirm setzen,

PTST

ermittelt die Farbe eines Punktes,

ALINE

zeichnet eine Linie zwischen zwei Punkten (entspricht dem herkömmlichen Befehl LINE). und außerdem können noch verschiedene Einstellungen vorgenommen werden. Alle bisher verwendeten Grafikbefehle können Sie jedoch weiterhin verwenden.

Der Vorteil dieser LINE-A Routinen liegt in der Geschwindigkeitssteigerung gegenüber den herkömmlichen Befehlen. Sämtliche Gem-Funktionen greifen übrigens auch auf die LINA-A Routinen zurück.

ENDIF
ENDPROC
ENDREPEAT
ENDSELECT
ENDSUB
ENDWHILE
EOF
EQV
ERASE
ERR$
ERROR
ERR
EVEN
EVERY
EVERY CONT
EVERY STOP
EVNT_BUTTON
EVNT_DCLICK
EVNT_KEYBD
EVNT_MESAG
EVNT_MOUSE
EVNT_MULTI
EVNT_TIMER
EXEC
EXIST
EXIT IF
EXP

FALSE
FATAL
FGETDTA
FIELD
FILES
FILESELECT
FILL
FIX
FLOAT
FN
FOR
FORM INPUT
FORM INPUT AS
FORM_ALERT
FORM_BUTTON
FORM_CENTER
FORM_DIAL
FORM_DO
FORM_ERROR
FORM_KEYBD
FRAC
FRE
FSEL_INPUT
FSETDTA
FSFIRST
FSNEXT
FULLW
FUNCTION

GB
GCONTRL
GDOS?
GEMDOS
GEMSYS
GET
GINTIN
GINTOUT
GOSUB
GOTO
GRAF_DRAGBOX
GRAF_GROWBOX
GRAF_HANDLE
GRAF_MKSTATE
GRAF_MOUSE
GRAF_MOVEBOX
GRAF_RUBBERBOX
GRAF_SHRINKBOX
GRAF_SLIDEBOX
GRAF_WATCHBOX

GRAPHMODE	HARDCOPY	HEX$	HIDEM
HIMEM	HLINE	HTAB	IF
IMP	INC	INFOW	INKEY$
INLINE	INP	INPAUX$	INPMID$
INPUT	INPUT$	INSERT	INSTR
INT	INTIN	INTOUT	KEYDEF
KEYGET	KEYLOOK	KEYPAD	KEYPRESS
KEYTEST	KILL	L:	LEFT$
LEN	LET	LINE	LINE INPUT
LIST	LLIST	LOAD	LOC
LOCAL	LOCATE	LOF	LOG
LOG10	LONG	LOOP	LOOP UNTIL
LOOP WHILE	LPEEK	LPOKE	LPOS
LPRINT	LSET	L~A	MALLOC
MAX	MENU	MENU KILL	MENU OFF
MENU_BAR	MENU_ICHECK	MENU_IENABLE	MENU_REGISTER
MENU_TEXT	MENU_TNORMAL	MFREE	MID$
MIN	MKD$	MKDIR	MKF$
MKIMKL	MKS$	MOD	MODE
MONITOR	MOUSE	MOUSEK	MOUSEX
MOUSEY	MSHRINK	MUL	NAME AS
NEW	NEXT	NOT	OBJC_ADD
OBJC_CHANGE	OBJC_DELETE	OBJC_DRAW	OBJC_EDIT
OBJC_FIND	OBJC_OFFSET	OBJC_ORDER	OB_ADR
OB_FLAGS	OB_H	OB_HEAD	OB_NEXT
OB_SPEC	OB_STATE	OB_TAIL	OB_TYPE
OB_W	OB_X	OB_Y	OCT$
ODD	ON x GOSUB	ON BREAK CONT	ON BREAK
ON BREAK GOSUB	ON ERROR	ON ERROR GOSUB	ON MENU
ON MENU BUTTON	ON MENU GOSUB	ON MENU IBOX	ON MENU KEY
ON MENU MESSAGE	ON MENU OBOX	OPEN	OPENW
OPTION BASE	OR	OTHERWISE	OUT
OUT?	PAUSE	PBOX	PCIRCLE
PEEK	PELLIPSE	PI	PLOT
POINT	POKE	POLYFILL	POLYLINE
POLYMARK	POS	PRBOX	PRED
PRINT	PROCEDURE	PSAVE	PSET
PTSIN	PTSOUT	PTST	PUT
QSORT	QUIT	RAD	RAND
RANDOM	RANDOMIZE	RBOX	RCALL
RC_COPY	RC_INTERSECT	READ	RECALL
RECORD	RELSEEK	REM	RENAME AS
REPEAT	RESERVE	RESTORE	RESUME
RETURN	RIGHT$	RINSTR	RMDIR
RND	ROL	ROR	ROUND
RSET	RSRC_FREE	RSRC_GADDR	RSRC_LOAD
RSRC_OBFIX	RSRC_SADDR	RUN	SAVE
SCRP_READ	SCRP_WRITE	SDPOKE	SEEK

Die Maus

Die Mausposition kann nun softwaremäßig geändert werden. Damit ist es möglich, eine Bewegung der Maus zu simulieren.

Mathematische Funktionen

Hier hat sich bei den trigonometrischen Funktionen einiges getan. Implementiert wurden die Funktionen

ASIN,
ACOS und
ATAN,

um den Arcussinus, Arcuscosinus und Arcustangens zu berechnen. Um einen Winkel vom Bogen- ins Gradmaß und umgekehrt zu berechnen, wurden zwei weitere Umrechnungsfunktionen implementiert. Dies sind

RAD und
DEG.

Die Funktionen

SINQ und
COSQ

liefern Sinus- und Cosinuswerte, die intern auf Grad-Werte in



- Der Hardvirus -







SELECT	SETCOLOR	SETDRAW	SETMOUSE
SERTTIME	SGET	SGN	SHEL_ENVRN
SHEL_FIND	SHEL_GET	SHEL_PUT	SHEL_READ
SHEL_WRITE	SHL	SHOWM	SHR
SIN	SINGLE	SINQ	SLPOKE
SOUND	SPACE$	SPC	SPOKE
SPRITE	SPUT	SQR	SSORT
STEP	STICK	STOP	STORE
STR$	STRIG	STRING$	SUB
SUCC	SWAP	SYSTEM	TAB
TAN	TEXT	THEN	TIMES
TIMER	TITLEW	TOPW	TOUCH
TRACE$	TRIM$	TROFF	TRON
TRUE	TRUNC	TYPE	UNTIL
UPPER$	USING	V:	VAL
VARPTR	VAR	VDIBASE	VDISYS
VOID	VQT_EXTENT	VQT_NAME	VST_LOAD_FONTS
VST_UNLOAD_FONTS		VSYNC	VTAB
V_CLRWK	V_CLSVWK	V_CLSWK	V_OPNVWK
V_OPNWK	V_UPDWK	V~H	W:
WAVE	WEND	WHILE	WINDTAB
WIND_CALC	WIND_CLOSE	WIND_CREATE	WIND_DELETE
WIND_FIND	WIND_GET	WIND_OPEN	WIND_SET
WIND_UPDATE	WORD	WORK_OUT	WRITE
W_HAND	W_INDEX	XBIOS	XOR
/	^	{}	~

Einige Befehle können durch Nachstellen von (und) einen anderen Sinn bekommen.

einer Tabelle abgelegt sind. Die Zwischenwerte werden in 1/16-Grad-Schritten linear interpoliert.

Dadurch werden Grafiken, die auf zeitaufwendigen mathematischen Verfahren basieren, ungleich schneller erstellt .Dabei ist eine Geschwindigkeitssteigerung um den Faktor 10 durchaus möglich.

GEM sei Dank

Zu guter letzt folgen noch die AES-Aufrufe des Betriebssystems. Diese wurden komplett implementiert.

Was soll man dazu noch sagen: Sämtliche Aufrufe des GEM können ab sofort als Befehl aufgerufen werden. Es sind keinerlei PEEKs und POKEs und keine externe Library nötig.

Das Buch der Bücher

Ein Schlußwort noch zu dem sehr umfangreichen Handbuch: Es handelt sich dabei um ein ca. 3.5 cm starkes Ringbuch mit weißem Papier und einer aus stabiler Pappe bestehenden schwarzen Hülle. Das Handbuch ist in zwölf nach inhaltlichen Gesichtspunkten geordnete Kapitel unterteilt, in denen zu Beginn jeweils eine kurze Zusammenfassung steht, der dann die einzelnen Befehle folgen. Zu jedem Befehl gibt es eine ausreichende Erklärung und jeweils mindestens ein Beispiel bildet den Abschluß.

Mit persönlich hat das zwölfte Kapitel gefallen am besten gefallen. Hier sind alle

GEMDOS-
BIOS- und
XBIOS-

Aufrufe zu finden. Alle Funktionsaufrufe und ihre Parameter werden hier kurz beschrieben.

Es folgt eine Tabelle der LINE-A Variablen, des WORK_OUT Feldes und V_OPN(V)WK Eingabeparametern. Danach kommen die VT-52 Steuerzeichen. Weiter geht es schließlich mit den Tabellen für Scan-Codes, den ASCII-Werten sowie der Füllmuster und Linien von GEM. Das Kapitel wird, nach den Erklärungen der Fehlermeldungen von GfA, Editor, Bomben und TOS, mit einem Index abgeschlossen.

Kompatiblität

Die neue Version ist selbstverständlich zu den alten Versionen kombatibel. Ein mit GfA 2.xx erstelltes Programm wird mit

SAVE AS

abgespeichert und in die Version 3.0 mit

MERGE

eingeladen. Bei einem Versuch mit einem 180 kByte langem Programm gab es keinerlei Schwierigkeiten. Beim Laden mit MERGE wird die Zeilennummer der gerade eingeladenen Zeile angezeigt. Hier kann man sehen, daß das Programm noch am Einladen ist und man kann die noch verbleibende Zeit abschätzen.

Ergebnis

Mit GfA-BASIC V3.0 hat man den momentan stärksten BASIC--Interpreter, der für den Atari ST zu haben ist. Es ist wirklich fast alles Denkbare in diese Programmiersprache eingebaut. Hier wird es einem wirklich einfach gemacht, ein sauberes Programm zu schreiben.

Natürlich muß man auch einen Blick auf das Haus OMIKRON werfen. Es bleibt abzuwarten, was nun aus dieser Richtung kommt.

Der Compiler

Der Compiler V3.0 soll etwa im Dezember 1988 erscheinen. Die Möglichkeit der Ansteuerung ist im BASIC schon vorhanden. Jedoch bleibt abzuwarten, wann der Compiler V3.0 fehlerfrei zu haben sein wird. Hier hatte GfA einige Schwierigkeiten mit den älteren Versionen gehabt.

Zu beziehen ist GfA-BASIC V3.0 im Fachhandel oder bei GfA in Düsseldorf zu einem Preis von DM 199.-. Besitzer von älteren Versionen erhalten es entsprechend günstiger. jw

MASTER ST

Eine zu Unix und MS-Dos kompatible Oberfläche auf dem Atari

Version 5.0

Immer der gleiche Ärger - neue Rechner, neue Befehle für den gleichen Arbeitsvorgang. Bestes Beispiel hierfür ist ERASE, ERA, REMOVE, DELETE, DEL etc.

Oder ein anderes Übel: Fenster auf, nochmal Fenster auf, Mausklick über Mausklick, bevor man am Ziel ist und sein Proramm starten kann. Dabei genügt es doch im Prinzip, wenn ein Pfad zu dem Directory, wo das auszuführende Programm zu finden ist, angegeben wird. Nach Eingabe des Programmnamens wird das Programm nun gestartet. Oder aber lange Befehle, die man immer wieder über Tastatur eintippen muß, könnten auch abgekürzt werden.

Ist es da nicht angenehmer, sich zu einem Befehl sein eigenes Befehlswort, versehen mit diversen Optionen, zu deklarieren?

Master ST vereinigt nun einige Features der MS-Dos und der Unix Welt. Dabei können unter Master ST Gem wie Tos Anwendungen (mit und ohne Parameter) aufgerufen werden. Ebenso ist es möglich, Master ST in einer anderen Umgebung, z.B. einer Gem-Oberfläche, oder aus einem Interpreter für eine Programmiersprache heraus zu verwenden. Vielseitigkeit ist also angesagt, und auf einige der Features von Master ST möchte ich hier näher eingehen.

Umfangreicher Befehlsvorrat

Master ST arbeitet tastaturorientiert, man vermißt die Maus jedoch aufgrund der umfangreichen Filezugrifftechniken von Unix überhaupt nicht. Eine Befehlsübersicht entnehmen sie bitte dem Schaubild 1.

Zu den einzelnen Befehlen vielleicht ganz kurz was sie bedeuten:

CHMOD <filename> ändert den Status von Dateien,
COMP vergleicht zwei Binärdateien,
DIFF liefert die Unterschiede in zwei Textdateien,
DUMP dient zum Ascii bzw. Hexdump von Files,
FIND findet Strings in einem Textfile,
HISTORY zeigt die letzten 20 ausgeführten Befehle an,
KEY dient zum Belegen der Funktionstasten,
LOCK setzt das Laufwerk auf schreibgeschützt.

Lehrer eingebaut

Nach dem Start von Master ST befindet man sich in dem Ordner 'Tutor'. Hier kann sich der Anwender durch Eingabe von 'Tutor' oder 'Experte' vom System in die Thematik einführen lassen (online tutorial). Dem Newcomer wird die File- und Ordnertechnik erklärt, daß Anlegen von Subdirectories, daß Kopieren von Files und das Anbringen von symbolischen Links (das ist keine Kopie sondern nur ein Pfad zum Original).

Experten wird erläutert, wie man Master ST z.B. in seiner täglichen Arbeit einsetzt, es werden sogenannte Alias angelegt und es wird nochmals auf die Linktechnik eingegangen. Beide Kurse erläutern zunächst die Theorie und geben dann dem Anwender die Möglichkeit, die zuvor erklärten Befehle selbst auszuprobieren.

Sind die Kurse durchgearbeitet, so kann mit der Arbeit begonnen werden. Durch 'help master' erhält der Anwender allgemeine Hinweise zu dem System. In Verbindung mit einem Kommando erhält man eine ausführliche Anleitung zu diesem Befehl.

ALIAS	ASK	BEEP	BYE	CAR
CAT	CDR	CD	CHMOD	CHRONO
CLS	COMP	COPY	CP	DATE
DEL	DELETE	DIFF	DIR	DUMP
ELAPSED	ERA	ERASE	ERR	EVAL
EXIT	FGREP	FIND	FILES	FORMAT
FREE	GET	GOTO	GREP	HELP
HISTORY	IF	IIF	INVERSE	KBD
KEY	LIF	LIIF	LL	LN
LOCK	LS	MD	MEMDUMP	MKDIR
MORE	MOVE	MV	NORMAL	PACK
PAGE	PAUSE	PWD	QUIT	RD
REM	REN	RENAME	RESTORE	REWIND
RM	RMDIR	RS232	SAVE	SET
SET?	SHIFT	SHOW	SLEEP	SHRINK
STATUS	TIME	TOUCH	TYPE	UNALIAS
UNLINK	UNLOCK	UNPACK	UNSET	VERSION
ZCAT	L25	L50		

Help Alias

Mit dem Befehl 'Alias' kann einem Kommando oder einer Kommandosequenz ein eigener Name zugewiesen werden. Dies geschieht in der Form:

ALIAS <name> <Kommando>

Durch 'Alias' können also bestehende Kommandos mit einem anderen Name versehen werden. Diese können zusätzlich mit Parametern versehen werden, wodurch der Anwender einiges an Tipparbeit spart.

Wenn zum Beispiel der Benutzer zum Löschen einer Datei den Befehl 'RM' mit der Option -I gesetzt hat, kann er durch Verwendung von

ALIAS RM Rm -I

erreichen, daß die Option nun ständig angewendet wird.

Im obigen Beispiel ist es wichtig, daß das Kommando anders als das neue Alias geschrieben wird, da sonst nochmal versucht wird, daß erhaltene Kommando zu expandieren und das könnte zu fatalen Rekursionen führen, wie sie z.B. 'Alias A A' darstellt.

Einem Alias kann auch eine komplette Kommandosequenz zugewiesen werden. Dabei ist nur zu beachten, daß ein ';' durch ein vorangestelltes '\' quotiert werden muß. Fehlt bei der Definition z.B. der Parameter <name>, so wird eine Liste aller vorhandenen Alias ausgegeben.

Beispiel:

ALIAS rm RM -I
ALIAS time Time\;DATE
ALIAS ed c:\util\edit.ttp

Help Pack

Der Befehl Pack und Unpack dient zum komprimieren von beliebigen Files und dem Wiederherstellen des Originalzustandes. Der Vorteil einer Komprimierung von Dateien liegt auf der Hand: Es wird Speicherplatz auf externen Medien wie Harddisk oder Floppy gespart.

Die Befehlsformate zu diesem Kommando lauten:

PACK <quelle> <ziel>
UNPACK <quelle> <ziel>

Folgende Optionen sind im Zusammenhang mit diesem Komando möglich:

O: Überschreibe ein eventuell existierendes Files ohne nachzufragen.
N: Überschreibe ein eventuell existierendes File nicht
D: Überschreibe ein eventuell existierendes File nur, falls es älter als die Quelle ist.
M: Lösche nach erfolgreichem Pack/Unpack die Quelle.
T: Ziel erhält den gleichen Zeit- und Datumstempel wie die Quelle, z.B. für Backups.
I: Vor jeder Aktion wird der Filename angezeigt und nachgefragt, ob das File bearbeitet werden soll. Mögliche Antworten sind Y (ja), N (nein), Q (keine weiteren Files), A (alle weiteren Files).

<quelle> und <ziel> sind beliebige Filebezeichnungen. Ist die Quelle nur ein Pfad (A:\ oder A:\C_Files\, dann werden alle Files von dort bearbeitet. Enthält die Quelle Wildcards, dann werden alle passenden Files verarbeitet. In diesen Fällen darf das Ziel (falls vorhanden) nur ein Pfadname sein. Fehlt die Bezeichnung des Ziels, dann wird die Arbeitsdirectory als Ziel genomen.

Existiert das zu kopierende File bereits und ist keine der Optionen N und O gesetzt, so wird zunächst gefragt, ob das Ziel überschrieben werden soll. Jedes bearbeitete File wird mit seiner ursprünglichen und seiner erzeugten Größe protokolliert. Dies kann im Quiet-Modus jedoch unterdrückt werden.

Bei der Verwendung des Kommandos Pack/Unpack ist noch zu beachten, daß ein File nicht unbedingt kleiner wird. So kann es passieren, daß das angeblich gepackte File länger ist, als das ursprüngliche File. Dies geschieht durch eine für den Algorithmus ungünstige Zusammensetzung der Daten. Hier verzichtet man besser auf ein Packen.

Beispiel:

PACK T.C T.PAK
PACK *.PAS A:\PAK\

Besonders angenehm auffallend ist die Optionsvielfalt der Befehle, auch wenn der 'normale' Anwender diese niemals in vollem Umfang verwenden wird.

Zum Lieferumfang gehört ein 70 Seiten starkes Handbuch, in dem die Help-Texte des Programmes nochmals in schriftlicher Form fixiert sind.

Vermißt habe ich Beispiele zum Gebrauch der Batchdateien, sowie Anwendungen im Zusammenhang mit Tos oder Gem Programmen. Auch den Kurs für Anfänger und Fortgeschrittene könnte man z.B. mit einer Nummerntechnik so ausstatten, daß man gezielt auf Kapitel zugreifen kann. Im übrigen erklärt sich das Programm weitgehend selbst.

Ein wenig experimentieren mit dieser Fülle an Befehlen wird dem Anwender sicher noch viel Spaß bereiten.

Fazit

Eine gelungene Implementierung des Softwarehauses Naumann & Röder GbR. Da zudem das Programm ständig erweitert wird, kann man auf die nächste Version gespannt sein.

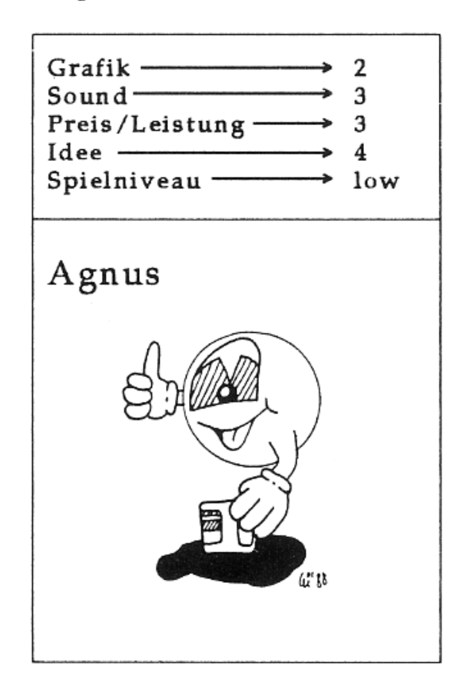
Bezugsquelle:
Naumann & Röder GbR
Am Sportplatz 22
6620 Völklingen

peba

THUNDERCATS

Hier ist Action pur angesagt, für Langeweiler kein Platz

Während unser Held Lion-O auf einer Dienstreise war, drangen die Molemen, Agenten des üblen Mumm-Ra, in den Cats-Lair ein, nahmen ein paar Mitglieder der Thunder-Cats mit



Grafik →	2
Sound →	3
Preis/Leistung →	3
Idee →	4
Spielniveau →	low

Agnus

und stahlen auch gleich das magische Auge der Thundera.

Lion-O schwört sich nun denn auch gleich, seine Freunde und den magischen Stein aus der Burg Mumm- Ra's zu befreien.

Ob Lion-O an Mumm-Ra's Gehilfen in den Wäldern und Höhlen vorbeikommen wird, liegt nun am Geschick des Spielers. Thundercats ist ein Spiel in Art von Barbarian von Psygnosis. Man läuft alleine durch Höhlen und andere hindernisreiche Gebiete und schießt oder schlägt auf seine Gegner ein (eine nicht sehr neue Spielidee) und erhält ab und zu

Thundercats, Elite Systems, England
20 engl. Pfund, ca. DM 60-80

die Gelegenheit, ein paar Gegenstände einzusammeln.

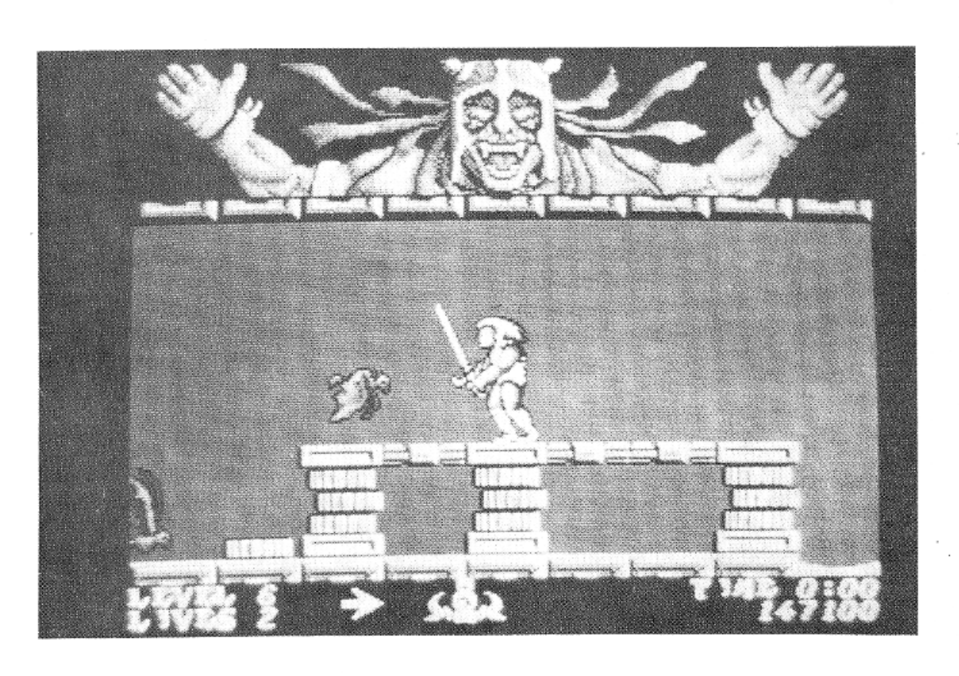
Trotz dem althergebrachten ersten Eindruck des Spiels, heldenhafte Story und wohlbekanntes Spielprinzip, kann man nicht sagen, daß es sich nicht lohnt ein Blick auf dieses Programm zu werfen. Es besitzt eine recht ansehnliche Grafik und ein gutes Scrolling. Der Sound ist zwar nicht die Krönung der musikalischen Schöpfung, ich habe aber schon weitaus schlechtere Sounds gehört.

Trotz der recht einfachen Steuerung des Kämpfers dauert es eine Weile, bis man sich auf einen 'grünen Zweig' gekämpft hat, was zum Teil an der allgemeinen Geschwindigkeit des Spieles, aber auch an den Scharen der Angreifer, die häufig in ganzen Rudeln auftreten, liegt.

Alles in allem ist das Spiel von dem bekannten Software-Haus Elite zwar keine absolute Neuerung, es ist aber trotzdem ein flottes Actionspiel mit dem man sich schon einige Zeit unterhalten kann.

Thundercats läuft nur auf Farbsystemen, dafür aber auch mit 512 Kb Speicher. Außer einer kopiergeschützen Diskettte liegt der Packung ein Faltblatt mit der Titelstory und den Ladeinstruktionen bei.

Der Preis von Thundercats beläuft sich in England auf 20 £. In Deutschland wird der Preis etwa DM 60-80 betragen. bm



ST VISION SONDERTEIL

OMIKRON BASIC

Neue Produkte, Tips und Trick und viele News

Inhalt:

MIDI MAZE EDITOR

Erstellen Sie eigene Labyrinthe für MIDI MAZE! Geschrieben in Omikron Basic

Wer kennt es nicht, das Multi-Player-Spiel Midi Maze? Bis zu 16 Spieler können in einem Labyrinth gegeneinander antreten und sich gegenseitig auflauern. Mit unserem Listing 'Midi Maze Editor' können sie sich jetzt eigene Spielfelder erstellen, natürlich Gem-gesteuert und in Omikron Basic, um eine optimale Mischung aus Anwenderfreundlichkeit und Geschwindigkeit zu erzielen.

Unser Listing besteht aus zwei Teilen. Dem Hauptprogramm und einem kleinen Hilfsprogramm, um die Resource-Datei zu erstellen. Wichtig ist, daß sie an das Hauptprogramm die Omikron Gem-Library anhängen. Diese ist schon im Lieferumfang des Omikron Basic enthalten und unter dem Namen 'GEMLIB.BAS' gespeichert. Nach dem Abtippen geben sie einfach 'MERGE GEMLIB' ein, speichern das Programm auf Diskette ab und schon stehen alle Gem-Funktionen zur Verfügung. Midi Maze Editor 1.0 läuft übrigens nur auf Farbsystemen.

Wer sich die Tipparbeit ersparen möchte, kann das fertige Programm auch über unseren PD-Service erstehen (siehe Seite 61-69).

Wir wünschen allen Anhängern von Midi Maze und Omikron Basic viel Spaß! rk/kuw

```
100 ' *************************************
110 ' ***  MIDI MAZE EDITOR Version 1.0  ***
120 ' ***   von Ralph Klepper (C) 1988   ***
130 ' *** in Omikron-Basic für ST Vision ***
140 ' *************************************
150 '
160 CLEAR
170 DEFINT "A-Z"
180 True=(0=0):False= NOT True:Resourcedefs
190 DIM C$(40)
200 Smem%L= MEMORY(27000)
210 Appl_Init
220 XBIOS (Rez,4)
230 IF Rez<>1 THEN
240    FORM_ALERT (1,"[3][Bitte auf
 mittlere|Auflösung umschalten][ Danke ]")
250    END
260 ENDIF
270 PALETTE $0,$770,$320,$707
280 Rsrc_Load("TEST.RSC",X)
290 IF X<>1 THEN
300   FORM_ALERT(1,"[3][RSC not found][ Abbruch ]")
310   Appl_Exit
320   END
330 ENDIF
340 Rsrc_Gaddr(0,Menu%L,Menu_Adr%L)
350 Menu_Bar(Menu_Adr%L)
360 Zeit2$= TIME$
370 Make
380 Wert=0
390 Wertx=5
400 Fill_A
410 FORM_ALERT (1,"[1]["+"Bitte die Midi Maze|
   Disc einlegen!"+"]["+" OK "+"]",C)
420 ON TIMER 5 GOSUB Time
430 '
440 REPEAT
450    Evnt_Mesag(X$)
460    FOR I=1 TO 7
470       Message(I)= CVI( MID$(X$,I*2+1,2))
480    NEXT
490    '
500    ' File
510    '
520    IF Message(4)=Laden THEN
530       Menu_Tnormal(File,1)
540       File_Laden
550    ENDIF
560    IF Message(4)=Speichern THEN
570       Menu_Tnormal(File,1)
580       File_Speichern
590    ENDIF
600    IF Message(4)=Quit THEN
610       Menu_Tnormal(File,1)
620       Ende
630    ENDIF
640    '
650    ' Labyrinth
660    '
670    IF Message(4)=Groesse THEN
680       Menu_Tnormal(Labyrinth,1)
690       Min_Max_Box
700    ENDIF
710    IF Message(4)=Leerbox THEN
720       Menu_Tnormal(Labyrinth,1)
730       Box_Paint
740       Im_Feld
750    ENDIF
760    '
770    ' Desk
780    '
790    IF Message(4)=About THEN
800       Menu_Tnormal(Desk,1)
810       FORM_ALERT (1,"[1][ Midi Maze
 Labyrinth Editor| by Ralph Klepper][ OK ]")
820       Menu_Tnormal(Desk,1)
830     ENDIF
840 UNTIL 0
850 '
860 DEF PROC Resourcedefs
870 '
880 Menu=0
890 Desk=3
900 File=4
910 Labyrinth=5
920 About=8
930 Laden=17
940 Speichern=18
950 Quit=20
960 Groesse=22
970 Leerbox=23
980 RETURN
990 '
1000 DEF FN Exist(Pfad$,File$)
1010 '
1020 LOCAL N,N$
```

```
1030 WHILE RIGHT$(Pfad$,1)<>"\"
1040    Pfad$= LEFT$(Pfad$, LEN(Pfad$)-1)
1050 WEND
1060 N$=Pfad$+File$
1070 IF N$<>"" THEN
1080    OPEN "F",2,N$,1
1090    GET 2,0
1100    N= NOT EOF(2)
1110    CLOSE 2
1120 ENDIF
1130 RETURN N
1140 '
1150 ' Einblenden der Uhrzeit
1160 '
1170-Time
1180 MOUSEOFF
1190 Zeit1$= TIME$
1200 IF Zeit1$<>Zeit2$ THEN
1210    Zeit2$= TIME$
1220    FILL COLOR =3: FILL STYLE =1,1
1230    PBOX 551,1 TO 638,9
1240    TEXT COLOR =1: TEXT STYLE =0
1250    TEXT 564,8, TIME$
1260 ENDIF
1270 '
1280 ' Atsch
1290 '
1300 Flag=Flag XOR 1
1310 IF Flag THEN
1320    FILL COLOR =2: FILL STYLE =1,1
1330    PELLIPSE 520,131,20,10,1800,0
1340    LINE COLOR =0
1350    DRAW 520,132 TO 520,139
1360 ELSE
1370    FILL COLOR =0: FILL STYLE =0,1
1380    PBOX 500,131 TO 540,134
1390    FILL COLOR =1: FILL STYLE =1,1
1400    PBOX 500,135 TO 540,141
1410 ENDIF
1420 MOUSEON
1430 RETURN
1440 '
1450 DEF PROC File_Laden
1460 '
1470 MOUSEOFF
1480 BITBLT 100,26,400,175 TO Smem%L
1490 MOUSEON
1500 REPEAT
1510    Pfad$="\*.MAZ"
1520    Name$=""
1530    FILESELECT (Pfad$,Name$,But)
1540 UNTIL But=0 OR LEN(Name$)<>0
1550 MOUSEOFF
1560 BITBLT Smem%L TO 100,26,400,175,3
1570 MOUSEON
1580 IF But THEN
1590    IF FN Exist(Pfad$,Name$) THEN
1600       A$=Name$
1610       OPEN "I",1,A$
1620       INPUT #1,B$,C$
1630       CLOSE 1
1640       A= LEN(C$)
1650       Wertx=A
1660       OPEN "I",1,A$
1670       FOR I=A TO 0 STEP -1
1680          INPUT #1,C$(I)
1690       NEXT I
1700       CLOSE
1710       Box_Paint
1720    ENDIF
1730 ENDIF
1740 RETURN
1750 '
1760 DEF PROC File_Speichern
1770 '
1780 MOUSEOFF
1790 BITBLT 100,25,400,170 TO Smem%L
1800 MOUSEON
1810 Pfad$="\*.MAZ"
1820 Name$=""
1830 REPEAT
1840    FILESELECT (Pfad$,Name$,But)
1850 UNTIL LEN(Name$)<>0 OR But=0
```

```
1860 MOUSEOFF
1870 BITBLT Smem%L TO 100,25,400,170,3
1880 MOUSEON
1890 IF But THEN
1900    OPEN "O",1,Name$
1910    A=Wertx
1920    FOR I=A TO 0 STEP -1
1930       PRINT #1,C$(I)
1940    NEXT I
1950    CLOSE
1960 ENDIF
1970 RETURN
1980 '
1990 DEF PROC Ende
2000 '
2010 FORM_ALERT (1,"[0]["+"Das Programm ist|
   hiermit beendet"+"]["+" OK | Nee "+"]",A)
2020 IF A=1 THEN
2030    Appl_Exit
2040    END
2050 ELSE
2060    FORM_ALERT (1,"[1]["+"OK Lassen wir|noch
   einmal|durchgehen !"+"]["+" OK "+"]",A)
2070 ENDIF
2080 RETURN
2090 '
2100 DEF PROC Min_Max_Box
2110 '
2120 REPEAT
2130    IF Wertx<5 THEN
2140       Wertx=5
2150    ENDIF
2160    Tw$="|OK|"
2170   Ta1$="Eigene Werte :| | Größe ="+ STR$(Wertx)+""
2180    Ta2$="| -           +":' 9 Spaces
2190    Ta$=Ta1$+Ta2$
2200    FORM_ALERT (2,"[0]["+Ta$+"]["+Tw$+"]",A)
2210    IF A=3 THEN
2220       Wertx=Wertx+2
2230       IF Wertx>33 THEN
2240          Wertx=Wertx-2
2250       ENDIF
2260    ENDIF
2270    IF A=1 THEN
2280       Wertx=Wertx-2
2290       IF Wertx<5 THEN
2300          Wertx=Wertx+2
2310       ENDIF
2320    ENDIF
2330 UNTIL A=2
2340 C$(0)= STR$(Wertx)
2350 Lang=Wertx
2360 Fill_A
2370 RETURN
2380 '
2390 DEF PROC Fill_A
2400 '
2410 Lang=Wertx
2420 FOR A=0 TO Lang-1
2430    C$(A)="."
2440    FOR B=1 TO Lang-1
2450       C$(A)=C$(A)+"."
2460    NEXT B
2470 NEXT A
2480 C$(Lang)= STR$(Lang-1)
2490 RETURN
2500 '
2510 DEF PROC Im_Feld
2520 '
2530 MODE =1
2540 FILL STYLE =1,1
2550 REPEAT
2560    Mx= MOUSEX
2570    My= MOUSEY
2580    Mb= MOUSEBUT
2590    B= INT(Mx/Feldx)
2600    A= INT((My-12)/Feldy)
2610    IF MOUSEBUT <>1 OR Olda<>A OR Oldb<>B THEN
2620       Wf=True
2630    ENDIF
2640    IF Wf=True AND Mb=1 AND A>=0 THEN
2650       IF B<Lang AND A<Lang THEN
2660          Wf=False
```

```
2670          Olda=A
2680          Oldb=B
2690          IF MID$(C$(A),B+1,1)="." THEN
2700             FILL COLOR =1
2710             MID$ (C$(A),B+1,1)="X"
2720          ELSE
2730             FILL COLOR =3
2740             MID$ (C$(A),B+1,1)="."
2750          ENDIF
2760       MOUSEOFF
2770       PBOX B*Feldx+1,A*Feldy+13 TO
    B*Feldx+Feldx-1,A*Feldy+Feldy+11
2780       MOUSEON
2790       ENDIF
2800    ENDIF
2810 UNTIL MOUSEBUT =2
2820 RETURN
2830 '
2840 DEF PROC Box_Paint
2850 '
2860 LOCAL Fy,Fy2,Fx,Fx2,Lf
2870 MOUSEOFF
2880 MODE =1
2890 FILL COLOR =0: FILL STYLE =1,1
2900 PBOX 0,11 TO 400,199
2910 Lang=Wertx
2920 Feldx= INT(399/Lang)
2930 Feldy= INT(Feldx*.472)
2940 Fy=Feldy:Fy2=Feldy+11
2950 Fx=Feldx:Fx2=Feldx-1
2960 LINE COLOR =2
2970 Lf=Lang*Feldx
2980 FOR X=0 TO Lang
2990    DRAW 0,X*Fy+12 TO Lf,X*Fy+12
3000 NEXT X
3010 Lf=Lang*Feldy
3020 FOR X=0 TO Lang
3030    DRAW X*Fx,12 TO X*Fx,Lf+12
3040 NEXT X
3050 Lf=Feldy+13
3060 FILL STYLE =1,1
3070 FOR A=0 TO Lang-1
3080    FOR B=0 TO Lang-1
3090       IF MID$(C$(A),B+1,1)="X" THEN
3100          FILL COLOR =1
3110       ELSE FILL COLOR =3
3120       ENDIF
3130      PBOX B*Fx+1,A*Fy+13 TO B*Fx+Fx2,A*Fy+Fy2
3140    NEXT B
3150 NEXT A
3160 MOUSEON
3170 RETURN
3180 '
3190 DEF PROC Make
3200 '
3210 MOUSEOFF
3220 FILL COLOR =2: FILL STYLE =1,1
3230 PRBOX 400,12 TO 639,31
3240 MODE =2
3250 TEXT COLOR =3: TEXT STYLE =20
3260 TEXT HEIGHT =8
3270 TEXT 407,25,"MIDI MAZE Labyrinth Editor",232,1,1
3280 TEXT COLOR =1: TEXT STYLE =4
3290 TEXT 407,25,"MIDI MAZE Labyrinth Editor",230,1,1
3300 MODE =1
3310 FILL COLOR =2: FILL STYLE =1,1
3320 PBOX 401,35 TO 639,199
3330 LINE COLOR =1
3340 BOX 401,35 TO 639,199
3350 MODE =3
3360 TEXT COLOR =2: TEXT STYLE =5
3370 TEXT HEIGHT =10
3380 TEXT 405,50,"MIDI MAZE EDITOR",230,1,1
3390 TEXT HEIGHT =6
3400 TEXT 405,60,"by the John-Player-Cop.",230,1,1
3410 MODE =1
3420 FILL COLOR =1: FILL STYLE =1,1
3430 PELLIPSE 520,109,110,45
3440 FILL COLOR =0
3450 PELLIPSE 520,89,110,45,2250,3150
3460 FILL COLOR =1
3470 PELLIPSE 520,89,110,40,2200,3200
3480 FILL COLOR =0
3490 PELLIPSE 480,94,20,15
3500 PELLIPSE 560,94,20,15
3510 FILL COLOR =2
3520 PELLIPSE 520,131,20,10,1800,0
3530 LINE COLOR =0
3540 DRAW 520,132 TO 520,139
3550 MODE =3
3560 TEXT COLOR =2: TEXT STYLE =4
3570 TEXT 405,169,"Ich wünsche euch viel",230,1,1
3580 TEXT 405,177,"Spaß beim erstellen",230,1,1
3590 TEXT 405,185,"neuer Labyrinthe",230,1,1
3600 MODE =1
3610 MOUSEON
3620 RETURN
3630 '
3640 ' MERGE "BASLIB" nicht vergessen
3650 ' um die Gem-Komponenten hinzuzufügen !
```

Die Resource-Datei

Nun folgt Teil 2 des Listings. Mit diesem kurzen Programmteil erstellen sie die für den Midi Maze Editor nötige Resource-Datei. In ihr befinden sich die Daten für die Menüleiste und der Inhalt der Pull-Down Menüs.

```
100 ' Programm zum Erstellen der
110 ' Resource-Datei für den
120 ' Midi Maze Editor
130 ' in Omikron Basic
140 ' ST Vision Nr. 7
150 '
160 CLEAR
170 DEFINT "A,W"
180 B= MEMORY(1000)
190 RESTORE
200 Anz_Soll%=856
210 Summe_Soll=41623
220 '
230 READ W%
240 WHILE W%<>-1
250    POKE B+Anz%,W%
260    Summe=Summe+W%
270    Anz%=Anz%+1
280    READ W%
290 WEND
300 '
310 IF Anz%<>Anz_Soll% THEN
320    PRINT "Falsche Anzahl von Datas"
330    PRINT "Sollwert: 856, gelesen: ";Anz%
340    END
350 ENDIF
360 IF Summe<>Summe_Soll THEN
370   PRINT "Prüfsummenfehler"
380   END
390 ENDIF
400 '
410 BSAVE "mme.rsc",B,856
420 '
430 END
440 '
450 ' Resource Datei
460 '
470 DATA 0,0,1,20,1,20,1,20,1,20,0,0,0,0,1,20,0,0
480 DATA 3,84,0,24,0,1,0,0,0,0,0,0,0,0,0,3,88
490 DATA 32,68,101,115,107,0,32,70,105,108,101,0,3
    2,76,97,98,121,114
500 DATA 105,110,116,104,0,32,32,69,100,105,116,
    111,114,0,45,45,45,45
510 DATA 45,45,45,45,45,45,45,45,45,45,45,45,45,45,
    45,0,32,32
520 DATA 68,101,115,107,32,97,99,99,101,115,
    115,111,114,121,35,49,0,32
530 DATA 32,68,101,115,107,32,97,99,99,101,115,115,
    111,114,121,35,50,0
540 DATA 32,32,68,101,115,107,32,97,99,99,101,
```

```
    115,115,111,114,121,35,51
550 DATA 0,32,32,68,101,115,107,32,97,99,99,101,
    115,115,111,114,121,35
560 DATA 52,0,32,32,68,101,115,107,32,97,99,99,
    101,115,115,111,114,121
570 DATA 35,53,0,32,32,68,101,115,107,32,97,99,99,
    101,115,115,111,114
580 DATA 121,35,54,0,32,32,76,97,100,101,110,0,32,
    32,83,112,101,105
590 DATA 99,104,101,114,110,0,45,45,45,45,45,45,
    45,45,45,45,45,45
600 DATA 0,32,32,81,117,105,116,0,32,32,65,117,
    115,109,97,158,101,0
610 DATA 32,32,76,97,98,121,114,105,110,116,104,
    32,98,101,97,114,98,101
620 DATA 105,116,101,110,0,255,255,255,0,1,0,6,0,
    25,0,0,0,0
630 DATA 0,0,0,0,0,0,0,0,0,80,0,25,0,6,0,2,0,2
640 DATA 0,20,0,0,0,0,0,0,17,0,0,0,0,0,0,80,2,1
650 DATA 0,1,0,3,0,5,0,25,0,0,0,0,0,0,0,0,0,2
660 DATA 0,0,0,23,3,1,0,4,255,255,255,255,0,32,0,0,0,0
670 DATA 0,0,0,36,0,0,0,0,0,6,3,1,0,5,255,255,255,255
680 DATA 0,32,0,0,0,0,0,0,0,42,0,6,0,0,0,6,3,1
690 DATA 0,2,255,255,255,255,0,32,0,0,0,0,0,0,0,
    48,0,12
700 DATA 0,0,0,11,3,1,0,0,0,7,0,21,0,25,0,0,0,0
710 DATA 0,0,0,0,0,0,3,1,0,80,0,19,0,16,0,8,0,15
720 DATA 0,20,0,0,0,0,0,255,17,0,0,2,0,0,0,19,0,8
730 DATA 0,9,255,255,255,255,0,28,0,0,0,0,0,0,
    0,59,0,0
740 DATA 0,0,0,19,0,1,0,10,255,255,255,255,0,
    28,0,0,0,8
```

```
750 DATA 0,0,0,68,0,0,0,1,0,19,0,1,0,11,
    255,255,255,255
760 DATA 0,28,0,0,0,0,0,0,0,88,0,0,0,2,0,19,0,1
770 DATA 0,12,255,255,255,255,0,28,0,0,0,0,0,0,0,
    107,0,0
780 DATA 0,3,0,19,0,1,0,13,255,255,255,255,0,
    28,0,0,0,0
790 DATA 0,0,0,126,0,0,0,4,0,19,0,1,0,14,
    255,255,255,255
800 DATA 0,28,0,0,0,0,0,0,0,145,0,0,0,5,0,19,0,1
810 DATA 0,15,255,255,255,255,0,28,0,0,0,0,0,0,0,
    164,0,0
820 DATA 0,6,0,19,0,1,0,7,255,255,255,255,0,
    28,0,0,0,0
830 DATA 0,0,0,183,0,0,0,7,0,19,0,1,0,21,0,17,0,20
840 DATA 0,20,0,0,0,0,0,255,17,0,0,8,0,0,0,12,0,4
850 DATA 0,18,255,255,255,255,0,28,0,0,0,0,0,0,0,
    202,0,0
860 DATA 0,0,0,12,0,1,0,19,255,255,255,255,0,
    28,0,0,0,0
870 DATA 0,0,0,210,0,0,0,1,0,12,0,1,0,20,
    255,255,255,255
880 DATA 0,28,0,0,0,8,0,0,0,222,0,0,0,2,0,12,0,1
890 DATA 0,16,255,255,255,255,0,28,0,0,0,0,0,0,0,
    235,0,0
900 DATA 0,3,0,12,0,1,0,6,0,22,0,23,0,20,0,0,0,0
910 DATA 0,255,17,0,0,14,0,0,0,23,0,2,0,23,
    255,255,255,255
920 DATA 0,28,0,0,0,0,0,0,0,242,0,0,0,0,0,23,0,1
930 DATA 0,21,255,255,255,255,0,28,0,0,0,0,0,0,0,
    252,0,0
940 DATA 0,1,0,23,0,1,0,0,1,20,-1
```

OMIKRON BASIC GEM-TUTOR

Programmieren sie in Omikron Basic? Haben sie sich bisher vor der GEM-Programmierung gedrückt?

Vom Hüthig Verlag Heidelberg ist nun ein guter Grund in Form einer Diskette erschienen, voll in die Gem-Programmierung mit Omikron-Basic einzusteigen. Wem die bisherige Omikron Gem-Library in ihrem Aufbau zu unübersichtlich war, bekommt hier ein Meisterwerk der Basic-Programmierkunst zu sehen.

Die Gem-Aufrufe der Omikron Library sind gerade für den Anfänger etwas verwirrend. Dazu kommt, daß ihre Implementation mit einigen Tricks geschah und auf eine ausführliche Dokumentation verzichtet wurde.

Es wurde platzsparend programmiert - in Basic bedeutet dies: Möglichst viele Befehle in eine Zeile stopfen. Leider wirkt dies auf viele Newcomer etwas abschreckend. Der Gem-Tutor dagegen ist 'sauber' programmiert und mit sehr vielen Kommentaren versehen.

Der Omikron Basic Gem-Tutor sieht sich aber nicht als umfassendes Gem-Nachschlagewerk, dazu reicht eine einseitig bespielte Diskette und ein 62 Seiten starkes Begleitheft nicht aus. Deshalb auch zu Beginn der Hinweis auf weiterführende Literatur zur Gem-Programmierung vom gleichen Autor.

Der Gem-Tutor stellt dem Anwender eine Vielzahl von nützlichen Funktionen und Routinen zur optimalen Ausnutzung der Gem-Komponenten zur Vefügung. Ob Dialogboxen, Drop-Down-Menüs oder die Eventverwaltung, zu jeder Problemstellung gibt es eine umfassende Programmroutine und erklärende Worte im Begleitheft. Sehr ausführlich wurde auch auf die Problematik der Fensterbehandlung eingegangen.

Die auf der Diskette befindlichen Routinen sind alle zum Einbau in eigene Programme vorgesehen und ersetzen die Omikron eigene Gem-Library völlig und besser!

Omikron-Basic Gem-Tutor ist erschienen im Hüthig-Verlag Heidelberg. Im Lieferumfang enthalten ist eine einseitig bespielte Diskette und ein 62 seitiges Begleitheft.

kuw

OMIKRON.BASIC 3.0

Der neue Basic Standard ist da - schnell, leistungsfähig und fast umsonst

Die Tage des alten ST Basic sind seit dem 01.07.88 gezählt - nun bekommt jeder Käufer eines Computers der Atari ST Serie kostenlos ein Omikron Basic Interpreter mitgeliefert. Aber nicht etwa die Version 2.0 sondern die erhebliche verbesserte Version 3.0.

Nicht weltweit

Der Entschluß von Atari, ein neues Basic in Umlauf zu bringen, gilt jedoch nicht weltweit. Das liegt daran, daß Atari nicht gleich Atari ist. So gibt es eigenständige Filialen für Deutschland, USA, England, Frankreich und Skandinavien. Franzosen, Engländer, Amerikaner und Schweizer haben leider noch das nachsehen, sie müssen weiterhin mit ST Basic auskommen.

In folgenden Ländern wird Omikron Basic 3.0 beim Kauf eines Ataris mitgeliefert:
Österreich, Yugoslawien, Niederlande, Belgien, Luxemburg, Dänemark, Schweden, Finnland und Norwegen und natürlich Deutschland, mittlerweile das Nummer 1 Land für den professionellen Atari ST Markt.

Für die nach Skandinavien ausgelieferten Interpreter gibt es noch eine Besonderheit. Um auch auf die landesüblichen Sonderzeichen nicht verzichten zu müssen, kann der Interpreter mit dem Befehl 'MODE SF' entsprechend konfiguriert werden.

Zur Sache

Das gesamte äußere Erscheinungsbild des Interpreters hat sich gewandelt. Der Full-Screen-Editor hat nun eine Menüleiste mit den Gem üblichen Pull-Down-Menüs. Jedoch wurde das Gem hier nur simuliert. Dies geschah aus Gründen der Absturzsicherheit und der Geschwindigkeit. Außerdem konnte hier gleich eine Funktion zum verlassen eines aktivierten Pull-Down-Menüs implementiert werden. Durch betätigen der rechten Maustaste verschwindet der heruntergeklappte Menüpunkt auf der Stelle.

Die Menüleiste

Um eine bessere Abgrenzung der Menüleiste zum Programmtext zu erreichen, erfolgt ihre Darstellung invers - für manche Gem-Programme sicher eine gute Idee, nur leider mit dem 'normalen' Gem nicht zu realisieren.

Die Menüleiste enthält die folgenden Einträge: File, Find, Block, Mode, Go, Run, die jeweilige X-Y Position des Cursors, die Länge des Programmes, wie es auf Diskette abgespeichert wird, sowie den Programm- bzw. Filenamen.

Nun könnte man sich leicht fragen, warum hier nicht eine Menüauswahl wie im Gfa-Basic erfolgte, bei der alle Funktionen direkt und ohne Pull-Down-Menüs zu aktivieren sind. Das Problem dabei ist, daß bei dieser Methode der halbe Bildschirm des Omikron Interpreters nur zur Funktionsauswahl belegt wäre.

Der neue Interpreter ist stark auf Funktionalität ausgerichtet. Auf den ersten Blick und vor allem ohne Handbuch, können seine Möglichkeiten kaum erkannt und genutzt werden.

Die Menüeinträge

Unter dem Eintrag File finden sich die üblichen Funktionen wie Load und Save, sowie zusätzlich die Möglichkeit zur Ausgabe der Directory mit den Fileattributen Länge und Datum.

Mehr zu bieten hat der Eintrag Find. Neben dem üblichen Suchen nach bestimmten Buchstabenfolgen, kann hier eine Suche nach Token eingeleitet werden. Diese Funktion ergab sich aus der Problemstellung, alle Variablen mit den Namen 'A' zu finden. Ein Find 'A' würde alle Wörter in denen ein A vorkommt anzeigen. Eine Suche mit Find Token würde nur das gewünschte Ergebnis, nämlich die Variable A, auflisten.

Das oben gesagte gilt natürlich auch für Prozeduren und Funktionen. Zusätzlich ist es möglich, durch Find Def direkt die Definition einer Prozedure oder Funktion zu suchen.

Variablennamen können mit Rename Token umbenannt und mit Print Token auf dem Drucker ausgegeben werden. Mit Print Token kann zusätzlich noch die Druckerbreite angegeben werden. Interessant ist auch die Tatsache, daß Einrückungen im Quelltext bei der Variablenausgabe auf dem Drucker übernommen werden.

Blinkende Fehler

Zur schnellen Erkennung von Syntax-Fehlern im Programmtext, wird die entsprechende Zeile durch blinken hervorgehoben, bis sich der Cursor in dieser Zeile befindet. Bei Bedarf kann diese Option abgeschaltet werden.

Durch Find Error können die fehlerhaften Zeilen direkt angesprungen werden, wobei der Cursor automatisch an der Stelle des Fehlers steht. Programme können übrigens trotz Fehler abgespeichert werden.

Blockfunktionen

Der Menüpunkt Block beschert uns die üblichen Funktionen wie move, delete, hide, print, load und save block. Positiv zu bewerten ist die Tatsache, daß Blöcke direkt mit der Maus ausgeschnitten werden können, hier muß sogar Tempus passen.

Ist der auszuschneidende Blockbereich größer als der Bildschirm, so wird automatisch im Listing nach oben oder unten gescrollt und die Zeilen als Block dargestellt, d.h. sie werden invertiert. Bemerkenswert auch hier wieder die Verarbeitungsgeschwindigkeit, die sich durchaus mit Werten des Tempus-Editors vergleichen läßt.

Interessante Funktionen liefert der Menupunkt Mode. Mit Change Size kann die Zeichengröße verändert werden. In der hochauflösenden Grafik kann zwischen 80 Spalten, 25 Zeichen und 104 Spalten, 44 Zeilen gewählt werden. In der Farbauflösung kann zwischen 80*22 und 40*22 Zeichen gewählt werden.

Splitting

Kenner der Version 2.0 werden sich noch an die Funktion des Bildschirmsplittens erinnern. Hierbei konnte der Bildschirm in zwei vertikale Bereiche aufgeteilt werden. Der Nachteil hierbei war, daß nur noch die Hälfte einer Zeile sichtbar war.

Die perfekte Lösung bietet nun das horizontale Splitting. Hierbei wird der Bildschirm ebenfalls in zwei voneinander unabhängige Bereiche unterteilt, wobei die Abgrenzung durch einen dicken Balken geschieht. Die Größe der Teilbereiche kann durch ein verschieben des Balkens frei gewählt werden. Ein wechseln des Teilbereiches geschieht durch anklicken mit dem Mauszeiger.

Mit Save Settings findet sich noch ein weiterer mächtiger Befehl in diesem Menüpunkt. Sämtliche im Interpreter und im Editor getätigte Voreinstellungen können in der Datei 'OMBASIC.INF' gespeichert werden.

Die sind im einzelnen die folgenden Punkte:

- Die Größe des verwendeten Fonts, Splitting und die Größe der Bildschirmbereiche
- Arbeiten im Insert- oder Replace-Modus, Zeilennummern an/aus
- Show Errors, also das Blinken fehlerhafter Zeilen
- Die Druckerbreite
- Der Fileextender, unter dem ein Blockbereich abgespeichert werden soll
- Definition der Variablenarten, wie DEFSNG, DEFDBL, usw.
- Belegung der Funktionstasten im Editor und Direktmodus sowie eine eventuelle Definition der rechten Maustaste

Was bietet der Menüpunkt Go? Hier können bis zu 4 Marken im Programmtext gesetzt und angesprungen werden. Leider ist diese Funktion nicht über Tastaturkommandos erreichbar.

Run

Vom Menüpunkt Run aus, kann ein Basicprogramm direkt vom Editor aus gestartet werden. Zusätzlich gibt es die Optionen Save&Run sowie Tron&Run.

Durch die Option Compile kann ein Basic-Compiler nachgeladen werden, der das Programm kompiliert. In der Compilerversion 3.0 kann dieser Vorgang direkt aus dem Arbeitsspeicher erfolgen. Bisher konnten nur auf der Diskette gespeicherte Programme kompiliert werden.

Mit Exec können wie vom Desktop aus beliebige Programme gestartet werden, ohne das der Interpreter verlassen werden muß. Diese Funktion ist natürlich auch vom Arbeitsspeicher des Computers abhängig. Anhänger von Desk-Accessories können nun auch aufatmen. Accessories können uneingeschränkt verwendet werden.

Soweit also zu den Funktionen des Editors. Das Basic selbst bekam 5 neue Befehle dazu:

Line Width: Bestimmung der Linienbreite für Draw-Befehle
Line Pattern: Definition eines Linienmusters
Fill Pattern: Definition eines Fillmusters
Text Rotation: Winkelangabe zur Textausrichtung
NDC: Umschalten von absoluten auf NDC-Koordinaten. Hierbei ist der Nullpunkt links-unten und die maximale Koordinate unabhängig von der gewählten Auflösung immer 32767,32767. Dadurch ist es möglich, Programme vollkommen unabhängig von der jeweiligen Auflösung zu schreiben, eine Umrechung und Anpassung der Koordinaten erfolgt automatisch.

Gesamteindruck

Nur wer sich mit den vielen Möglichkeiten des Interpreters einmal intensiver beschäftigt hat, kann all seine Funktionen würdigen. Im Vergleich zu Gfa 3.0 wirkt hier die Benutzeroberfläche auf den ersten Blick spartanisch, da nicht alle Möglichkeiten des Systems sichtbar sind. Die Vorzüge liegen im Detail und sind auf den Programmierer abgestimmt und nicht auf die Optik. So lassen sich Funktionstasten mit sämtlichen Befehlen, auch mit Mausklicks (!) belegen, so daß sich der Anwender neue Befehle selbst erstellen kann.

Für den professionellen Anwender, der auch ohne zusätzliche Schnörkel auskommt, gibt es eigentlich keine Alternative.

kuw

NEUE LIBRARIES FÜR OMIKRON-BASIC

Statistik, Numerik und Masken-Editor

Von der Firma OMIKRON sind mehrere Libraries erschienen, die wir Ihnen hier einmal vorstellen möchten.

Der Masken-Editor

Dieser bietet Ihnen die Möglichkeit, sämtliche Eingabemasken am Bildschirm zu entwerfen, Änderungen vorhandener Masken vorzunehmen und die fertigen Masken in eigenen Programmen einzubinden.

Zunächst wird der Editor gestartet. Nach Eingabe eines Dateinamens können beliebige Texte auf den Bildschirm geschrieben werden, wie sie im fertigen Programm erscheinen sollen. Dabei können Blöcke definiert, vervielfältigt und verschoben werden. Weiter können Eingabebereiche definiert werden, in denen im fertigen Programm der Text eingegeben wird. Dabei sind wie in einem Dateiverwaltungsprogramm verschiedene Eingabetypen möglich (Text, numerisch, Datum und Uhrzeit).

Überaus erfreulich sind zusätzliche Plausibilitätsprüfungen, ob z. B. eine Postleitzahl wirklich größer als 999 und kleiner als 9000 ist. 10 solcher Überprüfungen pro Eingabefeld können vorgenommen werden, danach kann man eine beliebige Fehlermeldung ausgeben oder sogar in eine Procedur springen, die dann anhand der Eingabe weiter Rechnungen vollführt, etc.

Wurde die Maske fertiggestellt, werden die Daten abgespeichert. Dabei erstellt der Editor 3 Dateien: die Definitionen von Konstanten, die PRINT-Befehle für die Maske und die Plausibilitätskontrollen, falls diese vorhanden sind. Diese Dateien werden dann mit zwei weiteren Eingabe-Routinen zum endgültigen Programm zusammengesetzt.

Zusätzlich sind mehrere Programme, z. B. eine Dateiverwaltung oder ein Biorhythmusprogramm beigefügt, in denen man die Funktionen und Möglichkeiten einmal testen kann, sowie Hilfsfunktionen, um die Eingaben weiter zu verarbeiten.

Die Numerik-Library

Sie ist mehr für mathematisch Interessierte und enthält eine Vielzahl von Möglichkeiten, naturwissenschaftlich-technische Probleme zu lösen.

Dies sind z. B. das Lösen von Gleichungen und Gleichungssystemen, verschiedene Approximationen, so auch die Fourier-Analyse und Fouriertransformation. Weiterhin ist es möglich, eine numerische Integration und Differentation durchzuführen.
Ebenfalls enthalten:
Lösen von Differentialgleichungen, Berechnung von Eigenvektoren und -werten.

Insgesamt stehen meist mehrere Funktionen zur Verfügung, die ein Problem auf verschiedene Weise lösen können.

Weitere spezielle Funktionen, die nicht im OMIKRON-BASIC implementiert sind, wurden nachgebildet. Als Stichwort sind hier nur die Gamma-, Beta- und Besselsche-Funktionen genannt.

Auch in dieser Library wurden Demoprogramme mitgeliefert, die die Benutzung der Funktionen verdeutlichen. Dabei ist auch ein Programm, um Funktionen auf dem Bildschirm zu zeichnen, das ebenfalls in eigene Programme eingebunden werden kann.

Die Statistik-Library

sei als letztes noch genannt. Sie enthält über 80 Funktionen, die sich den Problemen der Statistik widmet.

Dies sind Funktionen, um verschiedene Mittelwerte, Standardabweichungen, Varianzen etc. zu berechnen, dazu kommen noch die Verteilungen:
Standardnormalverteilung und ihre Umkehrfunktion, allgemeine Normal-, Exponential- und Binomialverteilung und vieles, vieles mehr.

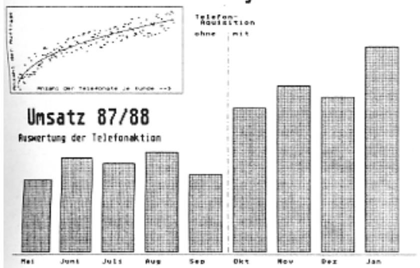
Für Simulationen sind eigene Zufallsgeneratoren vorhanden, die Werte verschiedener Verteilungen liefern. Weitere Funktionen beschäftigen sich mit Tests, Regression und Varianzanalyse. Hilfsproceduren für Grafikausgaben sind ebenso vorhanden, wie die Berechnung der Binomialkoeffizienten.

Ebenfalls in der Programmbibliothek dabei sind Demoprogramme, die den Umgang dieser Funktionen verdeutlichen.

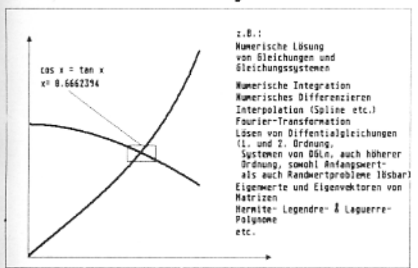
Alle Libraries können gleichzeitig in einem Programm eingebunden werden. Sie bieten eine nützliche Erweiterung der schon im BASIC implementierten Funktionen.

jw

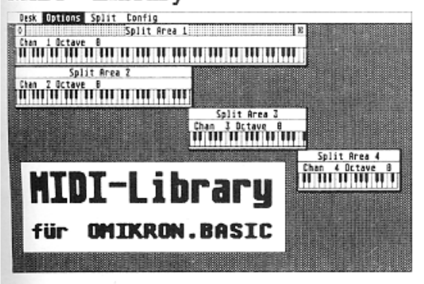
STATISTIK – Library



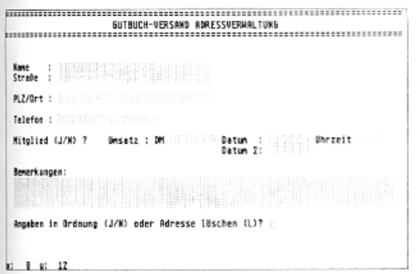
NUMERIK – Library



MIDI – Library



MASKENEDITOR



D.R.A.W.!





OMIKRON.BASIC: das Entwicklungssystem

„Der OMIKRON.BASIC-Interpreter stellt zusammen mit dem Compiler das beste BASIC-Entwicklungssystem für den Atari ST dar."
(ST Computer 9/87, S. 64)

Der Interpreter

Ein umfangreicher Befehlssatz, ein leistungsstarker Editor und höchste Ausführungsgeschwindigkeit kennzeichnen den OMIKRON.Interpreter.

- 19 Stellen Genauigkeit bei allen Funktionen
- Matrizen-Algebra
- Masken-INPUT, SORT-Befehl (mit deutschen Umlauten)
- Strukturiertes Programmieren: rekursive Prozeduren, mehrzeilige Funktionen
- 99 % MBASIC-kompatibel

OMIKRON.BASIC auf Modul

* Omikron Basic auf Diskette ist neuer Basic Standard auf Atari ST und liegt seit dem 1.07.88 jedem ST schon ab Werk bei.

Der Compiler

Noch mehr Tempo erreichen Ihre in OMIKRON.BASIC geschriebenen Programme durch unseren OMIKRON.Compiler.
Der Compiler erzeugt ein Assemblerprogramm (echte Maschinensprache, keinen P-Code!), das direkt vom Desktop aus angeklickt werden kann. Auch Accessories können Sie mit dem OMIKRON.Compiler erstellen. Durch CUTLIB werden nur diejenigen Ausführungsroutinen eingebunden, die tatsächlich gebraucht werden; die Folge sind außerordentlich kompakte Programme. Die Ablaufgeschwindigkeit der compilierten Programme ist außergewöhnlich. Der Tester in der mc meinte gar: **„Nur Assembler ist schneller"** (mc 2/87, S. 122)

OMIKRON.Compiler
OMIKRON.Compiler für 68881
(vollkompatibel zum normalen Compiler)

Die Libraries

Wer professionell Programme entwickelt, möchte nicht jedesmal das Rad neu erfinden. Mit neuen Libraries (Befehls-Bibliotheken) und Hilfsprogrammen macht OMIKRON.BASIC Ihnen das Leben jetzt noch etwas leichter. Libraries erweitern den Befehlssatz: Sie laden einfach die Library, und die neuen Befehle stehen Ihnen sofort zur Verfügung.
Außer den links abgebildeten Programmen gibt's noch eine Robot-Library (ab Mai) und einen **I**ntegrierten **D**ebugger **E**ditor **A**ssembler **L**inker (kurz **IDEAL**), mit dem Sie Maschinenprogramme erstellen und austesten können (ST Magazin 19, Seite 74: „Debugger der Superklasse")

Alle links abgebildeten Programme
IDEAL

Gerne schicken wir Ihnen nähere Informationen zu – Anruf genügt!



Programme für Programmierer

OMIKRON.SOFTWARE
Erlachstraße 15 · D-7534 Birkenfeld · ☎ (0 70 82) 53 86

FENSTER IN OMIKRON

Erklärungen zur GEM-Demo

In unsere letzten Ausgabe haben wir Ihnen gezeigt, wie Sie Menüleisten in Ihr Programm einbinden können. Neben Menüs sind jedoch noch Fenster von großer Bedeutung, um ein Programm bedienungsfreundlich und übersichtlich zu gestalten. Mit auf der OMIKRON-BASIC-Diskette befindet sich ein Demoprogramm zur Darstellung von Menüs und Fenstern mit dem Namen GEMDEMO. Wir möchten Ihnen dazu noch einige Tips geben, um dieses Beispielprogramm auch für Ihre eigenen Zwecke zu nutzen. Wir beschäftigen uns hierbei nur mit den Funktionsaufrufen, die sich mit der Fensterverwaltung befassen. Dies wäre als erstes

Wind_Get(0,4,X0,Y0,W0,H0).

Diesem Befehl wird die Nummer des Fensters (0) und ein weiterer Parameter (4) übergeben, anhand denen man verschiedene Daten zurückbekommt (in X0, Y0, W0 und H0). Mit 4 erhält man die Größe des Arbeitsbereiches des Fensters. Da es sich um das 0-te Fenster, welches immer unsichtbar im Hintergrund liegt, handelt, erhält man die Koordinaten des nutzbaren Bildschirmbereiches, also ohne die Menüleiste. Mit

Wind_Create($FFF,X0,Y0,W0, H0,Window)

übergeben Sie GEM die Parameter wie Fenstergrößen und nutzbare Komponenten (Schließfeld, Schieber etc.), die dabei bitweise verschlüsselt sind (siehe Tabelle). Wird das jeweilige Bit gesetzt, erfolgt eine Darstellung der betreffenden Komponenten. In unserem Fall wurden alle Bits gesetzt ($FFF). Weiterhin müssen Sie noch die Strings, die den Namen und den Titel des Fensters enthalten, übergeben und darstellen. Dies übernimmt

Wind_Set(Window,2,Text$, Adresse).

Window ist hier wieder die Nummer des Fensters, mit der 2 sprechen Sie den Namen des Fensters an. Es folgt der String und die Adresse eines Speicherplatzes, an der Freiraum für das Gemdos ist.

Mit einer 3 anstatt der 2 greifen Sie auf den Titel des Fensters zu.

Damit sind die Vorbereitungen für die Fensterverwaltung abgeschlossen.

Evnt_Mesag(X$) ist auch hier für die Abfrage aller eintreffenden Ereignisse zuständig. Diese werden dann für das bessere Verständnis in ein Array mit dem Namen Message() eingelesen. Die Abfrage

IF Message(0)=Mn_Selected

befaßt sich mit der Auswahl der Menüs. Hier müssen Sie beachten, daß beim Beenden des Programmes, die Fenster wieder gelöscht werden.

Bit Nr.	Typ	Bedeutung
0	Name	Titelzeile
1	Close	Löschfeld
2	Full	Bildschirmformatfeld
3	Move	Bewegungsfeld
4	Info	Informationszeile
5	Size	Formatfeld
6	Uparrow	Pfeil nach oben
7	Dnarrow	Pfeil nach unten
8	Vslide	vertikaler Schieber
9	Lfarrow	Pfeil nach links
10	Rtarrow	Pfeil nach rechts
11	Hslide	horizontaler Schieber

Die weiteren Hauptabfragen befassen sich mit:
dem Anklicken eines nicht aktiven Fensters (=Wm_Topped), den vier Pfeilen und den Scrollfeldern (=Wm_Arrowed), dem vertikalen und dem horizontalen Schiebern (=Wm_Hslid und Wm_Vslide).

Es folgt die Änderung der Größe (=Wm_Sized) und die Bewegung eines Fensters (=Wm_Moved), danach die Vergrößerung auf Bildschirmgröße (=Wm_Fulled) und das Löschen eines Fensters (=Wm_Closed). Die in Klammern angegebenen Namen sind Variablen, die einen Wert enthalten, der in der Gem-Library definiert wurde.

Um die einzelnen Fenster zu restaurieren, wird bei

IF Message(0)=Wm_Redraw

verzweigt. Hier muß der Programmierer für den Bildaufbau aller zerstörten Fensterteile sorgen. Es wird zunächst die Fenstergröße ermittelt und die Rechteckliste bestimmt. Diese wird erstmalig mit

Wind_Get(Window,11,X,Y,W,H)

aufgerufen und übergibt die Koordinaten der Rechtecke, die neu gezeichnet werden müssen. Die folgenden Rechteckkoordinaten werden mit derselben Funktion bestimmt, statt 11 schreibt man jedoch 12. In der Procedure Redraw erfolgt der Aufbau der eigentlichen Fenster. Durch die Funktion

Clip X,Y,W,H

wird die Grafikausgabe auf das entsprechende Rechteck begrenzt.

jw

MIDI IN DER PRAXIS

Ein Listing in C zur Aufteilung eines Keyboards in vier Bereiche

Wie in der letzten Ausgabe versprochen, bringe ich heute einen Beitrag zur MIDI-Programmierung. Bevor ich jedoch "im luftleeren Raum" über die Programmierung philosophiere, bringe ich Euch lieber eine Anwendung, die auch ein hör- und sichtbares Ergebnis liefert.

Das Programm CZ_Split macht es möglich, ein MIDI-Keyboard (vorzüglich CASIO) in bis zu vier Bereiche einzuteilen, die jeweils einen auswählbaren Klang ansteuern. Es ist in (Megamax-) C geschrieben, kann aber ohne Schwierigkeiten in BASIC umgesetzt werden. Vor der Besprechung des Listing ist jedoch noch etwas Theorie nötig.

Zur Wiederholung: Der ATARI ST und das MIDI-Instrument kommunizieren jeweils über die Ports MIDI_In und MIDI_Out miteinander. Der Anschlag des c' auf dem MIDI-Keyboard z.B. erzeugt neben dem (meist) hörbaren Ton aus dem (hoffentlich) angeschlossenen Verstärker eine Nachricht die über MIDI_Out des MIDI_Boards in den MIDI_In Port des STs. Eine Nachricht besteht laut MIDI-Definition aus einem Status-Byte und einer Reihe von Daten-Bytes. In unserem Fall erhält der ST das Status-Byte 144 ($90) und die Daten-Bytes 60 und 64.

ALLE Statusbytes sind erkennbar am gesetzten 7.Bit, sind also größergleich 128 ($80). Umgekehrt haben demnach Daten-Byte Werte zwischen 0 und 127. Das Status-Byte 144, auch als NOTE_On Event genannt, ist die Nachricht, daß ein Ton angeschlagen wurde. Das nächste Byte definiert die Tonhöhe, hier eine 60 für c'. Für jeder Wert zwischen 0 und 127 beschreibt (theoretisch) einen Halbton. Jedoch wer hat ein schon ein Keyboard mit 128 Tasten ?!? Es folgen die Daten für die Tonhöhe und die Anschlagsdynamik (Velocity), also praktisch die Lautstärke. (0 = stumm , 127 = höchste Lautstärke).

Halt - wer hat überhaupt gesagt, daß die Taste wieder losgelassen werden soll, denn dies ist ja wieder ein Ereignis, das mit einer entsprechenden Nachricht, der Bytefolge Note_Off 128 ($80) und wieder 60, gefolgt von einer 0 quittiert wird. Die '0' zum Schluß entspricht der "Loslaßgeschwindigkeit" der Taste - wirklich !

Nur die wenigstes Synthis können diesen Effekt jedoch nutzen. Zudem hält sich nicht jeder Synthi an diese Definition für "Ton aus" und sendet stattdessen ein Note_On mit der Lautstärke 0. Der Effekt ist derselbe. Jeder Ton sollte nach dem Anschlag auch wieder einmal ausgeschaltet werden - oder ?!? Neben den Nachrichten über die Tastatur können - je nach Ausstattung des Synthis - Informationen über Veränderungen verschiedener Steuerungseinrichtungen, wie Programmtasten, Portamento- oder Vibrato-Einstellungen weitergeleitet bzw. erzeugt werden.

Theoretisch können bis zu sechzehn verschiedene MIDI-Geräte über je einen MIDI-Kanal gleichzeitig angesteuert werden. Zu den oben stehenden Erklärungen muß ich also hinzufügen, daß z.B. die Bytefolge 144,69,64 nur für den 1. Kanal bedeutet, daß ein a' erklingen soll. Für Kanal 4 z.B. muß es 147,69,64 lauten. Allgemein gilt: zu jedem Status-Byte ist die Kanal-Nummer minus eins hinzuzuzählen. Normalerweise ist ja zu erwarten, daß beim Anschlag einer Taste auch irgenwo ein Ton erzeugt wird. Dank MIDI hört auch das auf. Der Befehl Local_Off schaltet alle angeschlossenen Tonerzeugungsblöcke auf stumm.

Wozu ? Nun, die über die Tastatur angespielten Töne werden nach wie vor als Nachrichten über den MIDI_Out Port ausgegeben, können also z.B. vom ST weiterverarbeitet werden, während er selbst den Soundblock Eures Synthis zu seinen eigenen Zwecken mißbraucht...

Die Modus-Befehle sind die letzte Befehlsgruppe, deren Anwendung ich jetzt noch erklären muß: mono_on oder poly_on bestimmen, ob der Synthesizer ein- oder mehrstimmig angesteuert wird. Omni_on und omni_off legen fest, ob eine Kanaltrennung vorgenommen wird oder nicht. Mit omni_off ist es dem Synthesizer nicht egal, auf welchem Kanal er seine Informationen bekommt, sondern er "hört" nur auf Daten, die auf "seinem" Kanal kommen, andere Kanäle ignoriert er. In dem CZ_Split-Programm wird Mono_On- Omni_Off verwendet, d.h. der Synthi wird einstimmig angesteuert mit bis zu vier aufeinanderfolgenden Kanälen, die die Stimmen für die zu definierenden Keyboard-Bereiche ansteuern.

So - nun geht's aber in die Praxis der Programmierung. Die MIDI-Ports werden beim ATARI ST über BIOS-Funktionen angesprochen; und zwar über den Kanal 3. Die vier Funktionen auf die es hier hauptsächlich ankommt sind im Listing in den Zeilen 50...100 zu finden. Bconstat(3) oder BIOS(1,3) fragt ab, ob vom MIDI_In Port ein

Byte empfangen werden kann.
Bcostat(3) oder BIOS(8,3) zeigt an, ob ein Byte ausgegeben werden kann.
Bconin(3) oder BIOS(2,3) übernimmt ein Byte vom MIDI Port (falls vorhanden).
Bconout(3,X) oder BIOS(3,3,X) sendet das Byte X über den MIDI Port.

Auf diese vier Funktionen baut nun der gesamte MIDI-Datenverkehr auf. Midi_clear() (Zeilen 100-102) z.B. sorgt nur dafür, vor der eigentlichen Datenübertragung eventuell noch "wilde" Datenbytes vor dem MIDI_In Port des STs abzufangen, die ansonsten zu Datensalat führen könnten. Das Listing ist (so hoffe ich) ausreichend dokumentiert, so daß weitere Erklärungen überflüssig werden.
Leider wurde das Listing umfangreicher als geplant, so daß wir den zweiten Teil in der nächsten ST VISION veröffentlichen. Unsere PD-Library enthält das komplette Listing jedoch schon! ws

```
/*      CZ SPLIT                                          */
/*                                                        */
/*      Wolfgang Schmidt, ST VISION        6/88           */
/*                                                        */
/***                                                    ***/
/*      CZ SPLIT teilt das Keyboard in bis                */
/*      zu vier Bereiche                                  */
/*      auf, die mit verschiedenen Klängen                */
/*      belegt werden können.                             */
/***                                                    ***/

#include <stdio.h>
#include <osbind.h>

/* BIOS Kanal-Nummer für MIDI                             */
#define MIDI              3

/* System-IDs der verschiedenen Hersteller               */
#define ID_CASIO          68
#define ID_Yamaha         67
#define ID_Korg           66
#define SYS_ID            ID_CASIO
/* z.B. für einen CASIO CZ-101  *

/* Channel-Voice Messages */
#define NOTE_ON           144
#define NOTE_OFF          128
#define CONTROL_CHANGE    176
#define PROGRAM_CHANGE    192

#define TRUE              1
#define FALSE             0
#define cls() printf("\33E");  /* Clear_Screen */

/* aktueller MIDI-Kanal (1..16)                           */
int midi_chan = 1;

int vprg[3], sp[3];
int mc, czt, i, nsplit, b, t, sp1, sp2, sp3;
int s;
char a;


/***            Die MIDI-Funktionen                     ***/

/* midi_in Status */
/* Status-Abfrage (lesen) des MIDI-Kanals.                */
/* -1 = ein Byte kann eingelesen werden                   */
/*  0 = kein Byte vorhanden                               */

int midi_in_stat()
{
        return(Bconstat(MIDI));
}
/* midi_out Status */
/* Abfrage, ob ein Zeichen an den MIDI-Kanal              */
/* ausgegeben werden kann                                 */

int midi_out_stat()
{
        return((int)Bcostat(MIDI));
}


/* MIDI-Input */
/* Einlesen eines Zeichens vom MIDI-Kanal.                */
/* (Wartet auf das Zeichen).                              */

int midi_in()
{
        return(0xFF & Bconin(MIDI));
}
/* MIDI-Output */
/* Ausgabe eines Zeichens an den MIDI-Kanal       */
/* Mit Rückmeldung über den Erfolg der Ausgabe.*/
int midi_out(d)
int d;
{
int r;
        if (midi_out_stat())
           {
           Bconout(MIDI,d);
           r = TRUE;
           }
         else
           r = FALSE;

        return(r);
}

/* MIDI-Daten 'auslaufen lassen' -                        */
/* Kanal frei lesen                                       */

midi_clear()
{
        while (midi_in_stat())
           midi_in();
}

/* Alle Töne des aktuellen Kanals ausschalten     */

all_notes_off()
{ midi_out(175+midi_chan);
  midi_out(123);
  midi_out(0);
}

/* Ab hier sind alle Routinen fest                        */
/* auf MIDI-Kanal 1 eingestellt                           */
/* Unterbrechung der Kopplung                             */
/* Keyboard <-> Klangerzeugung                            */

local_off()
{ midi_out(CONTROL_CHANGE);
  midi_out(122);
  midi_out(0); /* off */
}

/* Verbindung der Funktionen von                          */
/* Keyboard <-> Klangerzeugung                            */

local_on()
{ midi_out(CONTROL_CHANGE);
  midi_out(122);
  midi_out(127); /* on */
}

/* Omni-Modus ein */

omni_on()
{ midi_out(CONTROL_CHANGE);
  midi_out(125);
```

```
  midi_out(0);
}

/* Omni-Modus aus */

omni_off()
{ midi_out(CONTROL_CHANGE);
  midi_out(124);
  midi_out(0);
}

/* Poly-Modus ein */
poly_on()
{ midi_out(CONTROL_CHANGE);
  midi_out(127);
  midi_out(0);
}

/* Mono-Modus ein */

mono_on()
{ midi_out(CONTROL_CHANGE);
  midi_out(126);
  midi_out(8);
}

main()
{
int ia, ie;
  do {
  /* Init der Split-Points und Patchnummern-Felder */
     for (i=0; i<=3; i++)
        {
        sp[i] = 36;
        vprg[i] = 0;
        }
     cl_s();

    /* Abfrage des zu benutzenden MIDI-Kanals */
     do {
        printf("Gib den MIDI-Basis Kanal ein\
              (1..13): ");
        scanf("%d", &mc);
        }
     while( !check_range(1,13,mc));
            mc--;
     midi_chan = mc;

     /* Markenbezeichnung anfragen */
     puts("");
     puts("Welche Bezeichnung hat dein CZ\
           Synthi ?  ");
     puts("  1 - CZ-101/1000");
     puts("  2 - CZ-3000/5000");
     puts("  3 - andere (CZ-230S/CZ-1");
     do {
        printf("Welche Bezeichnung\
              (1, 2 oder 3): ");
        scanf("%d", &czt);
        }
     while (!check_range(1,3,czt));
     cl_s();

/* Einteilung der Keyboard-Bereiche in          */
/* verschiedene Klangfarben                     */
/* Klang 1: linkes Ende bis erster Split-Point,*/
/* Klang 2: ersten Split-Point bis zum Zweiten,*/
/* usw.                                         */

     do {
        printf("Wieviele Split-Points möchtest\
               du (1..3)?  ");
        scanf("%d", &nsplit);
        }
     while (!check_range(1,3,nsplit));
     for (i=1; i <= nsplit; i++)
        {
        do {
           printf("   \nGib den MIDI-Wert der\
                  Taste ein, auf der der\n");
           printf("   %d. Split-Point liegen\
                  soll (36..96): ", i);
           scanf("%d", &sp[i-1]);
           }
        while (!check_range(36,96,sp[i-1]));
        }
     sort();      /* Reihenfolge der Split-Points
                     absteigend sortieren */
     cl_s();

/* Auswahl der Klänge für d. Keyboard-Bereiche */
/* Falls die Cartridge angewählt wird und sie  */
/* ist nicht vorhanden wird der entsprechende  */
/* Klang in der Default-Bank genommen          */

     for (i=0; i<=nsplit; i++)
        {
        ia = (i!=nsplit) ? sp[i] : 36;
        ie = (i!=0) ? sp[i-1]-1 : 95;
        printf("Wähle nun den Klang für den\
               Bereich %d (%d..%d) aus:\n",
             i+1, ia, ie);
        if (czt == 1)
          {
          do {
             printf("   Bank (Preset = 1,\
                Internal = 2, Catridge = 3): ");
             scanf("%d", &b);
             }
          while (!check_range(1,3,b));
          do {
             printf("   Programm Nummer\
                   (1..16): ");
             scanf("%d", &t);
             }
          while (!check_range(1,16,t));
          }
        else if (czt == 2)
          {
          do {
             printf("   Source (Preset = 1,\
                   Memory = 2): ");
             scanf("%d", &b);
             }
          while (!check_range(1,2,b));
          do {
             printf("   Bank (A..D):");
             scanf("%c", &s);
             s = s - 'A' + 1;
             }
          while (!check_range(1,4,s));
          do {
             printf("   Programm Nummer\
                   (1..8): ");
             scanf("%d", &t);
             }
          while (!check_range(1,8,t));
          t = 8 * (s-1) + t;
          }
        else
          {
          b = 1;
          printf("  Programm-Nummer: ");
          scanf("%d", &t);
          }
        puts("");
        vprg[i] = 32*(b-1)+t-1;
     }
     sp1 = sp[0];
     sp2 = sp[1];
     sp3 = sp[2];
     mono_set();       /* setze mono-mode */
     cl_s();

     /* Daten zum Prüfen nochmals ausgeben */
     puts("Split-Points bei:");
     for (i=0; i<nsplit; i++)
       printf("%d.  %d\n", i+1, sp[i]);

     puts("");
     puts("Keyboard-Bereiche:");
```

FORTSETZUNG IN DER NÄCHSTEN ST VISION

CZ_SPLIT ist auch in unserer PD-LIBRARY ERHÄLTLICH!

LÄNDER DIESER ERDE

Wie sie ihre Kenntnisse in der Geographie spielend auffrischen können

Länder dieser Erde ist, wie der Name schon vermuten läßt, ein modernes Lernprogramm für den Bereich Geographie. Ziel des Programmes ist es, sich die geographische Lage eines Staates zu merken.

Das Programm basiert auf einer Methode der Psychologin Vera F.Birkenbihl, die davon ausgeht, daß sich die besten Lernerfolge einstellen, wenn nicht nur Text, sondern auch viel Grafik als Lernquelle dient. Zur Verwirklichung dieser Methode ist der Computer besonders geeignet.

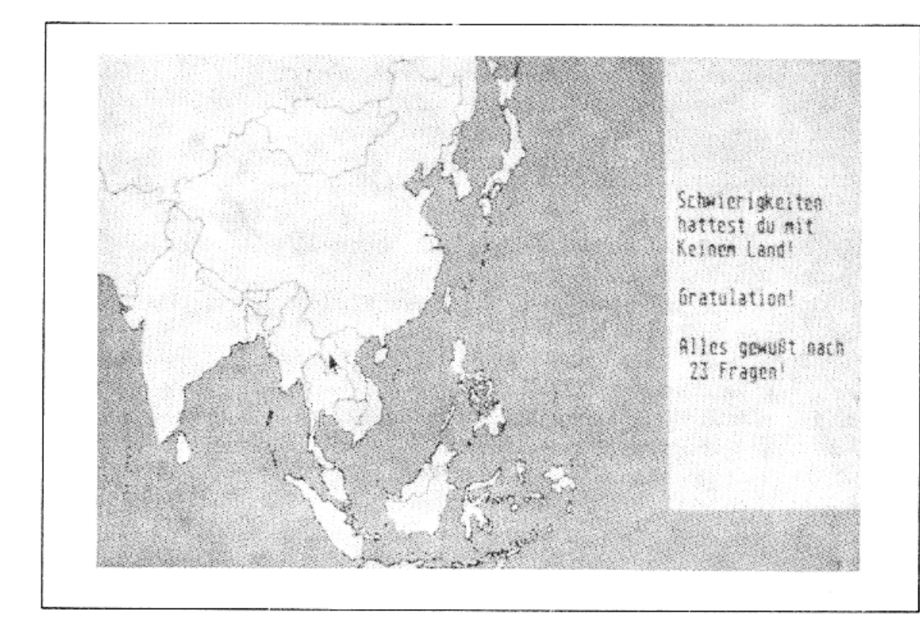
Bedienbarkeit

Nach dem Laden des Programmes wählt man durch anklicken eines Kontinents den Erdteil, über den man abgefragt werden möchte. Hat man dies getan, wird nun der ausgewählte Kontinent entsprechend vergrößert, damit er auf einer Bildschirmseite abgebildet werden kann. Der Computer fragt nun nach dem Namen eines Landes und man zeigt ihm dieses durch einfaches anklicken mit der Maus. Hier kann es manchmal zu Problemen kommen, nämlich dann, wenn die auszuwählenden Länder nur ein paar Pixel groß sind. Hier ist ein genaues Zielen mit Maus notwendig.

Genaue Kontrolle

Falls das ausgewählte Land richtig war, erscheint der nächste Name. War es aber das falsche Land, so zeigt der Computer die richtige Lage an. Sind alle Länder eines Kontinents einmal richtig ausgewählt worden, erstellt der Computer eine kurze Statistik.

Hier wird angegeben, wieviel Fragen man benötigt hat um einen Kontinent komplett zu erkennen. Außerdem werden alle Länder aufgezeigt, mit denen man Schwierigkeiten hatte. Diese werden schwarz ausgefüllt, so daß man unter Umständen nur noch schwarz sieht.



Burkina Faso

Das Programm erfüllt seinen Zweck, nämlich die Lage der Länder zu vermitteln, sehr gut. Wer vorher noch nicht wußte, was Burkina Faso ist, bekommt hier sogar die geographische Lage dieses Landes vermittelt.

Hauptstädte fehlen

Sind aber einmal alle Kontinente gelernt, bietet das Programm leider keinen weiteren Anreiz mehr. Ich hätte von 'Länder dieser Erde' mindestens noch die Frage nach der Hauptstadt erwartet. Hierdurch hätte man wirklich umfassende geographische Kenntnisse vermitteln können.

Fazit

Gerade für Kinder ist diese Art von Erdkunde hervorragend geeignet, so daß ich dieses Programm nur empfehlen kann.

Leider konnten wir den genauen Preis dieses in Omikron Basic geschriebenen Programmes nicht in Erfahrung bringen. Er wird aber unter DM 50.- liegen.

Zu beziehen ist 'Länder dieesr Erde' bei:

Lern Partner
Jahnstr. 9/1
7535 Königsbach-Stein

ATARI-ALLERLEI

Tatsachen, Gerüchte und Neuankündigungen aus den USA

Aufruhr bei Atari

Vollkommen unbemerkt vom europäischen Computermarkt fand in den USA bei Atari im Frühjahr 1988 eine komplette Neustrukturierung der Firma statt. Seit Mai existieren unter der Schirmherrschaft der Atari Corp. die drei Tochtergesellschaften Atari Computer, Federated und Entertainment Division. Die wohl für ST Besitzer interessanteste der drei neuen Firmen ist zweifelsohne Atari Computer. Und dort ging es in den ersten zwei Monaten drunter und drüber.

An die Spitze von Atari Computer wurde Chuck Babbit, ehemaliger Vize-Präsident des weltweit bekannten Softwarehauses Ashton Tate, gestellt. Tony Gould, der für zehn Jahre bei IBM im Verkauf tätig war, wurde zum Vize Präsident der Atari Computer Verkaufsabteilung ernannt. Die bisher in den USA eingesetzten Zwischenhändler wurden ausgeschaltet. Atari Computer verkauft seine Produkte jetzt über angestellte Handelsvertreter direkt an seine Händler. Ziel dieser Maßnahme ist es, die Atari Händler Kette zu verstärken. Die Federated Laden Kette hat im Rahmen dieser Umorganisation jetzt auch endlich die Mega ST Serie in seine Produktpalette aufgenommen.

Atari Computer suchte nun eine Werbeagentur, da Babbit für das dritte Quartal 1988 eine großangelegte Werbekampagne für die Mega ST Serie plante. Als diese Nachricht in den USA die Runde machte, hofften viele Atari Fans, daß dadurch endlich eine Wende in den USA eingeläutet würde. Denn der ST führt dort zur Zeit ein absolutes Schattendasein. Er konnte sich auf dem US Markt bisher kaum durchsetzen. Noch immer haftet Atari das Spiel Image an. Wen wundert dies, nachdem nur vor nicht allzu langer Zeit, das Atari 8-bit Computer System, das von seinen Besitzern über Jahre hinweg immer wieder als leistungsfähige Computer aufs heftigste verteidigt wurde, plötzlich von Atari selbst neu durchgestylt als die neue Spielmaschine angeboten wurde.

Doch Babbit konnte oder besser durfte, die Hoffnungen, die er erweckt hatte, nicht erfüllen. Mitte Juni - keine zwei Monate nach seinem Einstieg bei Atari - saß er auch schon wieder auf der Straße. Was war passiert ? Offensichtlich hatte er nicht das richtige Verständnis für die Tramiel'sche Weltanschauung.

Das Managment von Atari Corp. ließ zu seiner Entlassung verlauten, daß er versucht hätte, eine Werbekampagne in den USA zu starten, die die Firma tief in die roten Zahlen gestürzt hätte.

Parallel zu diesen Geschehnissen lud Atari die vierzehn besten Atari Händler nach Sunnyvale ein, um einen Tag mit den Verantwortlichen der Atari Corp. zu verbringen. Alle vierzehn Händler erschienen auch zum vereinbarten Termin. Eines der Hauptthemen dieses Tages war, wie verkaufe ich ein Produkt, daß ich überhaupt nicht oder nur in geringen Mengen auf Lager habe !!! Hierzu muß man wissen, daß die Händle in den USA zur Zeit, was Atari Produkte angeht, ziemlich auf dem Trockenen sitzen.

Durch den auch im Sommer immer noch anhaltenden RAM-Baustein Engpaß war Atari gezwungen Prioritäten zu setzen. Da der ST sich in Europa mittlerweile einen relativ guten Marktplatz erkämpft hat, ging ein Großteil der produzierten Geräte nach Europa und nicht in die USA. Insbesondere die Mega ST Serie konnte in den USA erst wesentlich später in den Geschäften bewundert werden als hier in Europa. Die Atari PCs sind in den USA zur Zeit noch überhaupt nicht erhältlich.

Nun aber zurück zu dem besagten Treffen in Sunnyvale: Atari unterbreitete seinen Händlern einige interessante Angebote. Meist handelte es sich dabei um Komplettpakete - angefangen von der XE-Spielmaschine über ein 520 ST-Lern-Paket (incl. SM124, einem Lernprogramm, Planetarium und Missile Command) für US$ 699, ein DTP Paket (Mega ST2, SM124, Laserdrucker SLM804, Microsoft Word, Vip Professional und VT-100 Cartridge) für US$ 2995 bis hin zum Geschäftsbüropaket (Mega ST4, SLM124, SLM804, die neue Harddisk MegaFile20 sowie die gleiche Software wie beim DTP-Paket zuzüglich dem TimeWorks Publisher) für US$ 3995. Zu den Preisen ist zu sagen, daß das letzte Paket, orientiert man sich an den bisher in den USA gültigen Listenpreisen, um US$ 1.600

günstiger ist als wenn man sich das System einzeln zusammenkauft.

Auf zukünftige Werbung angesprochen, ließen die Verantwortlichen verlauten, daß eine nationale Werbekampagne immer noch geplant sei. Allerdings wolle Atari auch durch regionale Werbung im Verbund mit den Händlern bei Kostenteilung die Verkäufe der Atari Produkte stimulieren...

Atari macht Schlagzeilen

Aber auch in positiver Hinsicht geriet Atari in diesem Sommer in die Schlagzeilen. Computer Shopper, eine amerikanische Compterzeitung in ungewöhnlich großem Format (größer als DIN A4), widmete Atari diesmal seine Titelseite und einen großen Leitartikel. Auf dem Cover prangte ein Photo des Atari DTP-Paketes mit dem Untertitel 'ATARI MEGA ST4 und LASERDRUCKER - ein kraftstrotzendes DTP-Duo'. Früher konnte man in diesem Magazin, Atari-Informationen mit der Lupe suchen.

Einen Monat vorher konnte man über die Mega Serie sogar etwas in dem renomierten Computer Magazin BYTE lesen.
Noch aussagekräftiger allerdings sind zwei Artikel in der bekannten US Zeitung Washington Post, die ebenso von Atari handelten. Dies könnten wirklich erste Anzeichen sein, daß Atari vielleicht doch auch in den USA allen Unkerufen zum Trotz mit Hilfe der Mega ST Serie als ernstzunehmender Computerhersteller auf sich aufmerksam macht. Ob diese guten Anzeichen allerdings der Beginn einer neuen hoffnungsvollen Ära für Atari USA sind oder nur ein kurzes Strohfeuer, dürfte sicherlich in nicht unwesentlicher Weise davon abhängen, ob Atari im Stande ist, die Lieferengpässe seiner Produkte abzubauen.

Neues aus der Atari Gerüchteküche

Das der Computermarkt sehr schnellebig ist, ist allgemein bekannt. Kaum ist ein Computer auf dem Markt, so wird in den Entwicklungslabors schon an seinem Nachfolger gebastelt. Insbesondere Atari ist ja dafür berühmt, auf Messen noch nicht produktionsreife Modelle vorzustellen oder anzukündigen, die dann meist sehr viel später - oder überhaupt nicht - auf den Markt kommen.

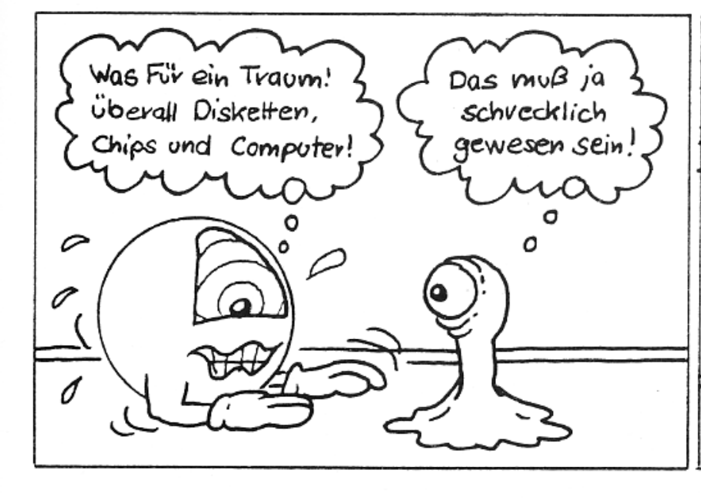
Zur Zeit ist die Gerüchteküche mal wieder ganz schön am brodeln. Hier nun ein paar neue (oder auch schon alte!) Informationen rund um Atari. Was davon Wirklichkeit wird, wird die Zeit uns weisen.

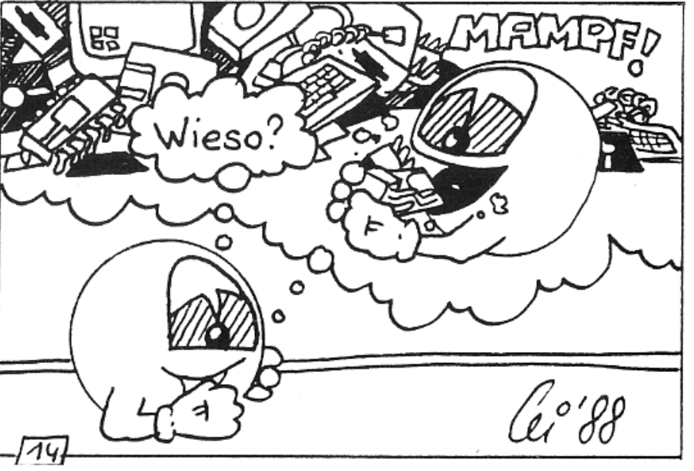
Noch in diesem Jahr soll der bereits mehrfach erwähnte und auf der CeBIT'88 noch durch Systemabstürze glänzende 68030 Computer erscheinen. Ebenso kurz vor der Veröffentlichung soll der erste Laptop ST mit eingebauter Harddisk stehen.

Daß Atari immer noch auf Videospiele setzt ist ebenso allegemein bekannt. Hier soll Nintendo nun der Kampf angesagt werden und zwar mit einer auf dem 68000er basierenden Spielmaschine für US$ 199. Der Atari Firmengründer Nolan Bushnell steht zur Zeit bei Atari unter Vertrag, mit der Aufgabe Spiele für dieses neue System zu entwickeln. Ob da im nächsten Jahr die XE-Spielkonsole etwa durch eine ST-Spielkonsole abgelöst werden soll ???

Auch vom Transputer wird noch geredet. Fünfzig Maschinen befinden sich bereits seit einiger Zeit in den Händen von Softwareentwicklern. Weitere 200 Entwicklungssysteme sollen diesen Sommer noch ausgeliefert werden. Für die Öffentlichkeit zugänglich soll der Transputer 'voraussichtlich' Anfang 1989 werden.

Man darf gespannt sein, was für weitere Überraschungen Jack Tramiel noch auf Lager hat. mts





BIT, BYTE UND BUS

Wir haben für Sie nachgeschaut: Grundbegriffe des Computers

Nicht nur Anfänger haben Probleme mit Fachausdrücken, auch fortgeschrittene Programmierer stoßen hin und wieder auf Schwierigkeiten. Wir haben für Sie die wichtigsten Begriffe erklärt. Haben Sie Fragen zu einem Thema, rufen Sie uns an. Wir greifen es gerne auf und berichten in der nächsten Ausgabe.

BIT

(Abkürzung für binary digit)
Bit stellt die kleinste Informationseinheit des Computers dar. Dabei können nur zwei Zustände auftreten: 1 und 0.

Binäres Zahlensystem, Dualsystem

Dies ist ein Zahlensystem auf der Basis der Zahl 2. Jede Ziffer kann nur die beiden Werte 0 oder 1 annehmen. Dies entspricht den von einem Computer darstellbaren Zustände 'an' oder 'aus', Plus- oder Minuspol einer Magnetzelle, usw. Jede Zahl des Dezimalen Zahlensystems läßt sich in eine binäre Zahl umrechnen. Wie beim dezimalen System entscheidet die Stellung einer Ziffer über ihren Wert, im dezimalen System das 10-fache, im binären System das 2-fache.

Neben dem binären Zahlensystem sind im Umgang mit Computern noch zwei weiter Zahlensysteme üblich. Dies sind das Oktal- und das Hexadezimalsystem (octo = lat. acht, hexadezi = gr.-lat. sechzehn).

Fast alle Computer rechnen mit dem binären Zahlensystem. Eine Ausnahme sind teilweise die Taschenrechner, in denen die Ziffern der Dezimalzahlen einzeln gespeichert werden. Jede Ziffer wird in vier Bit dargestellt, dem sogenannten BCD-Code. Somit treten beim Taschenrechner keine größeren Rundungsfehler auf. Der Nachteil ist jedoch der etwas größere Speicherbedarf und die wesentlich langsamere Geschwindigkeit aller Rechenarten.

Geben Sie einmal 1E-08 + 1 - 1 in den Computer und in den Taschenrechner ein. Hier sehen Sie den Vorteil dieses Codes: Der Taschenrechner gibt 1E-08 als Ergebnis aus, der Computer teilt 9.999993722E-09 als Ergebnis mit.

Doch zurück zum Dualsystem; es hat trotz seiner einfachen Rechengesetze einen großen Nachteil, der den Umgang erschwert: Schon relativ kleine Zahlen haben sehr viele Stellen.

Um diese Unübersichtlichkeit wenigstens für Maschinenprogrammierer zu vermindern, sucht man nach Möglichkeiten, andere Stellenwertsysteme zu verwenden, ohne die Vorteile des Dualsystems zu verlieren.

Oberstes Gebot dabei ist, daß die - immer notwendige - Umwandlung ins Dualsystem völlig durchsichtig ist und mit keinerlei Rechenarbeit verbunden sein darf.

Hierbei werden im Oktalsystem immer drei, im Hexadezimalsystem immer vier Ziffern des Dualsystems zusammengefaßt. Um drei Ziffern des Dualsystems zusammenzufassen, benötigt man 8 Ziffern, bei vier Ziffern des Dualsystems 16 Ziffern. (siehe auch untenstehende Tabelle).

Binär	Oktal	Hexadezimal	Dezimal	BCD-Code	ömische Ziffern
000000000000	0000	000	0	0000 0000 0000 0000	nicht möglich
000000000001	0001	001	1	0000 0000 0000 0001	I
000000000010	0002	002	2	0000 0000 0000 0010	II
000000000011	0003	003	3	0000 0000 0000 0011	III
000000000100	0004	004	4	0000 0000 0000 0100	IV
000000000101	0005	005	5	0000 0000 0000 0101	V
000000000110	0006	006	6	0000 0000 0000 0110	VI
000000000111	0007	007	7	0000 0000 0000 0111	VII
000000001000	0010	008	8	0000 0000 0000 1000	VIII
000000001001	0011	009	9	0000 0000 0000 1001	IX
000000001010	0012	00A	10	0000 0000 0001 0000	X
000000001011	0013	00B	11	0000 0000 0001 0001	XI
000000001100	0014	00C	12	0000 0000 0001 0010	XII
000000001101	0015	00D	13	0000 0000 0001 0011	XIII
000000001110	0016	00E	14	0000 0000 0001 0100	XIV
000000001111	0017	00F	15	0000 0000 0001 0101	XV
000000010000	0020	010	16	0000 0000 0001 0110	XVI
...					
000000011101	0035	01D	29	0000 0000 0010 1001	XXIX
000000011110	0036	01E	30	0000 0000 0011 0000	XXX
000000011111	0037	01F	31	0000 0000 0011 0001	XXXI
000000100000	0040	020	32	0000 0000 0011 0010	XXXII
000000100001	0041	021	33	0000 0000 0011 0011	XXXIII
...					
000101110011	0563	173	371	0000 0011 0111 0001	CCCLXXI
...					
111111111111	7777	FFF	4095	0100 0000 1001 0101	MMMMXCV
...					

Byte

Informationseinheit, bestehend aus 8 BITs und oft zusätzlich einem Sicherungsbit. Ein Byte ermöglicht die Verschlüsselung von 256 (entspricht 2 hoch 8) verschiedenen Zeichen, Buchstaben oder Zahlen. Ein Byte kann eine Zahl oder ein codiertes Zeichen enthalten.

KByte, MByte

Ein KByte sind genau 1024 Byte (1024 = 2 hoch 10). Die Abkürzung K kommt von Kilo = 1000. Man wählt hier die Näherung: 1024 Byte ~ 1 KByte ~ 1000 Byte. MByte entspricht einer Million Byte (M ist Abk. von Mega und entspricht einer Million). Auch hier gilt die Näherung: 1 Million Byte ~ 1048576 Byte (= 2 hoch 20).

Diese Angaben beziehen sich meist auf die Größe der Speicherkapazität von Computern, Disketten und Festplatten.

ROM

(read only memory = Nur Lese Speicher)
Speicher, der bei seiner Herstellung ein festes Programm erhalten hat, auf das der Anwender keinen Einfluß hat. Auf einem ROM können die Grundbefehle bzw. die Grundelemente eines Betriebssystems enthalten sein. Diese Funktionen stehen dann beim Einschalten des Computers zur Verfügung, ohne das sie von anderen Speichermedien in den Computer eingeladen werden müssen.

RAM

(random access memory=Schreib- und Lese-Speicher)
Die Arbeitsspeicher von Computern sind meist aus RAM-Bausteinen aufgebaut. RAM's sind flüchtige Speicher, d.h. bei Ausschalten der Spannungsversorgung verlieren sie den Speicherinhalt. Abhängig vom internen Aufbau unterscheidet man statische und dynamische RAM's. Dynamische RAM's speichern jedes Bit als Ladung eines Kondensators ab. Da sich die Kondensatoren langsam selbst entladen, muß die Ladung periodisch aufgefrischt werden (=refresh).

Dynamische RAM's sind relativ billige und schnelle Schreib/Lesespeicher, deshalb ihr Einsatz im Arbeitsspeicher.

PROM

(programmable read only memory - programmierbarer nur Lesespeicher)
Speicherbaustein, dessen Inhalt der Computer nur lesen, aber nicht verändern kann. Der Anwender kann diesen Baustein ein einziges mal mit einer speziellen Schaltung programmieren, danach können jedoch keine Änderungen mehr vorgenommen werden. Sind bei der Programmierung Fehler unterlaufen, muß ein neuer Baustein verwendet werden.

(E)EPROM

(electrically erasable programmable read only memory = (elektrisch) löschbarer nur Lesespeicher)

Im Unterschied zum PROM ist der Speicherinhalt wieder löschbar. Die abgespeicherten Daten sind nicht flüchtig, d.h. sie gehen bei Spannungsabfall (Ausschalten des Computers) nicht verloren. Die gespeicherten Daten sind nur mit Hilfe einer speziellen Schaltung oder eines Programmiergerätes änderbar.

Hauptspeicher, Arbeitsspeicher

Speicher, mit dem der Prozessor unmittelbar verkehrt. Er enthält Programme und Daten, die der Prozessor unmittelbar benötigt.

Selten benötigte Programme und Daten sind dagegen meist auf Plattenspeicher oder Diskette ausgelagert und werden nur bei Bedarf in den Arbeitsspeicher nachgeladen. Die Programme und Daten des Betriebssystem sind in der Regel immer im Hauptspeicher.

Der Arbeitsspeicher ist in kleine Speicherplätze unterteilt. Die Länge dieser Unterteilungen beträgt je nach Prozessor ein, zwei oer sogar vier Byte. Diese sind, um auf sie zugreifen zu können, durchnumeriert und daher einzeln beschreib- und lesbar.

Arbeitsspeicher sind meist aus Halbleiter - Speicherbausteinen aufgebaut, daher häufig die Bezeichnung RAM für den Arbeitsspeicher.

Betriebssystem

Gesamtheit der Programme, die den Betrieb eines Computers ermöglichen. Das Betriebssystem übernimmt die Kommunikation zwischen dem Prozessor und dem Anwender, verwaltet den Hauptspeicher sowie die angeschlossenen Peripheriegeräte (peripherie = gr.-lat. Rand, Randgebiet, hier: Geräte, die an den Computer angeschlossen sind: Monitor, Drucker, etc.).

Die bekanntesten Aufgaben des Betriebssystems sind die Verwaltung der Speichermedien (Hauptspeicher, Diskette) sowie die Ein- und Ausgabe von Daten. Hierzu ein kleines Beispiel: Das Warten auf eine Eingabe und Ausgabe des entsprechenden Zeichen auf dem Bildschirm.

Der Tastaturprozessor unter der Tastatur überwacht die Eingabe. Wurde eine Taste gedrückt, wird das laufende Programm unterbrochen und der Wert des entsprechenden Zeichens in den Hauptspeicher eingelesen. Es wird in eine weitere Routine des Betriebssystems verzweigt, die das Bitmuster des eingegebenen Zeichens auf dem Bildschirm entsprechend darstellt.

Dazu müssen jedoch noch die Koordinaten berechnet werden, an denen das Zeichen auf dem Bildschirm erscheinen soll. Es

wird daraufhin eine Adresse innerhalb des Bildschirmspeichers berechnet, in die das Bitmuster hineingeschrieben wird.

Die bekannten Betriebssysteme für PC's sind CP/M und MS-DOS. Das auf dem Atari ST eingesetzte Betriebssystem nennt sich TOS, was ursprünglich ein Akürzung für 'The Operating System' war. Kurz darauf hatte sich jedoch der Name 'Tramiel Operating System' eingebürgert, manche sprechen aber auch vom 'Terror Operating System'...

CPU, Mikroprozessor

Die CPU ist ein Prozessor, bestehend aus Steuerwerk, Rechenwerk und Spezialspeicher, der auf einem Chip untergebracht ist. Auf einem Chip können bis zu 300000 Schaltelemente (Transistoren) enthalten sein. Diese ProzessorEinheit ist sozusagen das Herz eines jeden Computers. Das Maß für ihre Leistungsfähigkeit ist die Anzahl der Bits, die die CPU gleichzeitig verarbeiten kann.

Am einfachsten läßt sich die CPU mit einem Verschiebebahnhof vergleichen, auf dem dauernd neue Datenzüge vom Laufwerk, Speicher oder von der Tastatur ankommen, hier neu zusammengestellt, bearbeitet oder wieder abgeschickt werden. Damit die Datenzüge nicht kollidieren, gibt es einen Fahrplan und eine vorgegebene Richtgeschwindigkeit, den Takt. Je schneller die CPU rangiert, desto leistungsfähiger ist sie. Für die Richtgeschwindigkeit und deren Einhaltung ist der Taktgenerator in der CPU verantwortlich. Sein Herzstück ist ein Quarz, der mit einer genau festgelegten Frequenz schwingt. Der im Atari ST eingesetzte Prozessor MC 68000 arbeitet mit einer 8 Megahertz Taktfrequenz.

BUS

Verbindung von Hardwareteilen über ein gemeinsames, von allen Geräten benutztes Leitungssystem. Der Vorteil liegt in der leichten Erweiterbarkeit durch zusätzliche Geräte oder Module.

Adressbus, Datenbus

Für die Übertragung von Adressen und deren Inhalte sind der Adress- und der Datenbus zuständig. Wenn Sie einen Wert im Speicher ablegen wollen, müssen Sie nicht nur den Wert kennen, sondern müssen auch wissen, wo dieser Wert hingeschrieben werden soll. In Programmiersprachen stehen Ihnen Variablen zur Verfügung. Dies sind aber nichts anderes als Platzhalter für eine Adresse. Der Befehl: LET A=17 bedeutet also, daß im Speicher ein Platz reserviert wird, in den der Wert 17 eingelesen wird. Mit A ist eigentlich die Adresse dieses Speicherplatzes gemeint.

Für die Übertragung von Daten bei Mikroprozessoren findet man oft die Bezeichnung 8, 16 oder 32 Bit-Prozessor (siehe CPU). Dies besagt, wieviele Datenleitungen der Datenbus umfaßt. Z. B. sind im inem 32 Bit-Prozessor 32 Datenleitungen vorhanden, es können also 32 Datenbits gleichzeitig vom Prozessor zum Hauptspeicher übertragen werden.

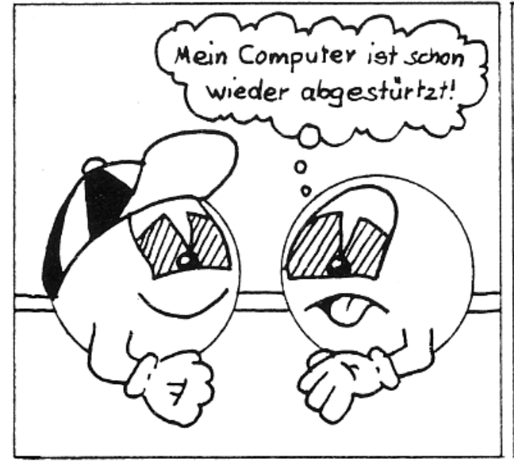
Hardware, Software

Ein Computer umfaßt grundsätzlich Hardware und Software. Unter Software versteht man alle auf einem Computer einsetzbaren Programme. Dies sind, angefangen vom Betriebssystem, sämtliche Anwenderprogramme und auch Spielprogramme. Zur Hardware gehört alles, was 'anfaßbar' ist, also alle mechanische und elektronische Teile.

eb



- Der Hardvirus -







PD-LISTE

Auszüge aus der aktuellen Library

PD-DISK:171

10x Spectrum 512 Bilder Format mit 512 Farben gleichzeitig! (f) 4Nudes - British - Composit - Jan - Phil2 - Redwing - Smooth - Spec05 - Spec06 - Spec10 - Spec11 - Teri - ZMagnum (Vorher Changhertz starten)

PD-DISK:172

Digitalisierter Sound von The Beatles "Let it be" 1=Laden / 2=Play / 3=Endlos-Play

PD-DISK:173

Digitalisierter Sound von Dire Strait " MTV " 1=Laden / 2=Play / 3=Endlos-Play

PD-DISK:174

Digitalisierter Sound von Blancmanche 1=Laden / 2=Play / 3=Endlos-Play

PD-DISK:175

ARCHIVSTAR 5.0: Archiviert Disketteninhaltsverzeichnisse

Degas - Compactor, -Decompactor

Hasedemo

KONV-ONE: Compactet und setzt Umlaute in DB-Master-One Reports (s/w)

Show Pic 2 mit Bild (f)

Ray-Animation: The magic bullets (s/w)

PD-DISK:176

Digitalisierter Sound von Michael Jackson " Bad "

PD-DISK:177

10x Spectrum 512 Bilder Format mit 512 Farben gleichzeitig (f) (Vorher Changehertz starten)

Spheres: Animation mit Bällen (f)

PD-DISK:178

Mouse Editor: mit ca. 100 fertigen Icon's

6x ACC's Directory - Lupe - Spool & Spooler - Tiny Tool - Watch

PD-DISK:179

SIC_PIC: Grafikeinbindung in Signum 1.0 (alle Bilder, alle Größen)

ZS_EDITOR24: Zeichensatzed. für 24Nadeldr. Erzeugt Down-Load (s/w)

MANAGER:ACC zum Einstellen eines NEC P5/6/7-Druckers

PRINT_OUT: Zum Ausdruck von Source-Code mit Schriftartwechsel

POSTER: Druckt Bild in vierfacher Größe (s/w)

INHALT: Erzeugt eine Index Liste von 1st Word-Texten

INDEX_PLUS: Erstellt Seitenindexlisten mit Lexikon

BT_KONVERT: Konvertiert 1st Word-Texte nach Beckertext

TYPEWRITER: Schreibmaschienenkurs mit über 40 Lektionen (s/w)

MORSE: Morsetrainingsprg. mit Übungen in Wort und Ton

PD-DISK:180

R_COPY: Backupprg. von Ramdisk auf Disk mit Source in Pascal

PASHELP: Crossreferenzprg. für Pascal, Ausgabe normal /Tempusfile

UNIDRUCK: Druckprg. mit Sonderzeichen (incl. Source in Pascal)

FONT: Zum Einbinden von Degas Fonts in Pascalprg. (Incl. Source)

BASICREF: Crossreferenzprg. für GFA-Basic (Incl. Source) (s/w)

MSK_EDIT: Maskeneditor für GFA-Basic (Incl. Source) (s/w)

RETTEN: Speichert zu lange Dateien in mehreren Teilen auf Disk's

SECO: Verwaltet Bildausschnitte und fügt sie zu Biblioteken zusam.

WORLD: Lernprg., fragt alle Länder und Hauptstätte der Erde ab. Mit Landkarten von der BRD und jedem einselnen Kontinent

PD-DISK:181

ZEITMANAGER: Semiprofessionelle Terminplanverwaltung, eigener Desk Top, erinnert an Termine, druckt Listen, verwaltet Timing, mit eingebautem Kalener

METROPOL: Wirtschaftsspiel. Sie müssen durch Manipulation div. Faktoren die Wirtschaft und Politik im Gleichgewicht halten

DALLAS: Hierbei dreht es sich um Öl. Sie als JR kämpfen um Profit und gegen Ihre Konkurrenten Alles in s/w

PD-DISK:182

SHERLOCK: Die Aufgabe ist es einen Mord aufzuklären. Sie schlüpfen in die Rolle des Sherlock Homes und können per Mausklick die Personen befragen

EL_BONZO_CITY: Ein Textadvanture mit interessantem Parser. In mittlerer Schwierigkeit und vielen Gags in der Story

SCRIBBLE: Eine schöne Version für ein oder zwei Spieler (s/w)

KREUZWORT: Programm zum Generieren eines Kreuzworträtsels (s/w)

UNITERM: Terminalprogramm mit Vielen Auswahlmöglichkeiten in GEM

PD-DISK:183

BRUCH: Umwandlung von Dezimalzahlen in Brüche. Listing in GFA und in Omikron Basic

DEG: Wandelt Degas-Bilder in das Screen-Format (32000 Bytes) und umgekehrt

DEMO: Grafikdemos in GFA-Basic

FILE: Programm, um Filenamen zu ändern. Nun sind auch Sonderzeichen möglich

OSTERN: Alle Ostertermine seit 1582, In Omikron-Basic

PLOTDEMO: Demoversion von ST-Plot

WEB: Globale Programmstrucktur zum Programmieren in Pascal, bestehend aus Editor, TANGLE, WALDI, WEAVE, Shell und Druckeranpassung, sowie Erklärung und Beispielprogramm.

PD-DISK:184

FUSSBALL: Sie sind der Manager einer Fußballmanschaft und sollen sie an die Spitze bringen (s/w)

HARDCOPY: Druckeranpassung für 24Nadel NEC-Drucker

LABEL: Super Programm zum bedrucken von Disketten-Etiketten

PD-DISK:185

Lamp's Demo (f)

PD-DISK:186

U_Emacs

Student CAD2: Objektorientiertes Grafikprogramm, es verwaltet alle Objekte, diese können ohne Auflösungsverlust vergrößert, verkleinert, gestaucht etc. werden und natürlich alle Standardfunktionen, so z.B. Box, Spiegeln, Text, Rotieren, Zoom etc. (s/w)

PD-DISK:187

AUTOCOPY: Copiert eine bestimmte Liste in beliebige Ordner

DATEBOOK: Erinnert beim Booten an wichtige Termine

PROFILRAMDISK: Resetfeste RAM Disk mit Ausführungsprofil

KONVERT: Konvertiert SM-Manager Texte in 1st Word

STATISTIK: Grafische Darstellung und Datenauswertung (s/w)

LABEL EXPERT: Druckt alle Art

von Etiketten (s/w)

VOKABELTRAINER: Zum Vokabellernen, mit schön gestalteter Eingabe

GRAFIK: 3D-Balkengrafik (s/w)

QUIZ: Eine Art Trivial Pursuite, eigene Fragen können eingegeben werden

VIDEO: Verwalten einer Videodatei unter GEM in Basic

CUBE HACK: Terminalprogramm mit integrierter Programmiersprache. Variablen, Kontrollstrukturen, Druckersteuerung, Rechenfunktionen, automatisches Einloggen in Mailboxen, etc.

OR SPREAD: Spreadsheetähnliches Programm zur Lineoptimierung, Netzplanetwicklung, Matrizen bis zu 500*500 (s/w)

PD-DISK:188

PLANET: Himmelskörperberechnungen, berechnet Ephemeriden aller heliozentrischer Himmelskörper (z.B. Mond, etc.). Anzeige aller Sterne und Sterndaten, und deren Animation.

TURING: Realisierung eines Turing-Maschinen-Modells (s/w)

ZUST: Analyse und Simulation linearer Regelkreise im Zustandsraum (s/w)

HORRORSCHLOß: Sie müssen ein Schloß erkunden, doch Geister und Falltüren behindern Sie

INVERSI: Das bekannte Spiel, bei dem Sie mehr Steine umdrehen müssen als Ihr Gegner (s/w)

INVATION: Space Invaders mit 3D Vektorgrafik (s/w)

ZARGE: Ein 32000Byte Bild wird in viele Teile geteilt, vermischt und muß nun wieder sortiert werden (s/w)

PD-DISK:189

ABC GEM Desktop 2.2
Graphs - Images - Wcharts

PD-DISK:190

Zu PD-Nr.189 ABC GEM 2.2
Draw - Graph - Paint - Wordchart (Mikroskop - Observat - Perspl 1 - Stapler)

PD-DISK:191

21 Digitalisierte Bilder von dem Film " Die Zeitmaschine "
TEIL 1 - mit ShowPic2 (f)

PD-DISK:192

17 Digitalisierte Bilder von dem Film " Die Zeitmaschine "
TEIL 2 - mit ShowPic2 (f)

PD-DISK:193

17 Digitalisierte Bilder von diversen Filmen, so z.B. 2* James Bond, 6* King Kong etc. - mit ShowPic2 (f)

PD-DISK:194

STWriter: Die Umsetzung des 8-Bit Textverarbeitungsprogramms als PD Version. Mit GEM-Einbindungen, so das es jetzt noch einfacher zu handhaben ist. Hat eine altbewährte Benutzeroberfläche die die 8-Bit User lieben gelernt haben.

PD-DISK:195

ANIMATE3: Brandneue Version (3.0) des RUN-ONLY-Programms für Cyber-Studio bzw. Cyber-Paint erstellt Kurzfilme und Animationssequenzen. Auf dieser PD-Disk die Super Animation von Oli " Fisch-, Glasdemo " (8 Sequenzen).

PD-Disk:196

Catch-Me: Fangen einer Alert-Box (Farbe) / Dumgem: Gem-Demo mit Modula Sourcecode / Gag-Tos: Grafikdemo / SerialEdit: Erzeugt zufällige Seriennummer für Disketten / Tempus PD / TGS: The Graphic Shell / Viren: Kleine Käfer krabbeln über den Screen / Virus-Killer / V-Killer 1 / V-Killer 2 / File-Fix.ACC: Neue Fileselector-Box von J.Sonander / Privat-Eye.ACC: v2.0 Statusanzeigen über Speicherplatz, usw. / Speedmus.ACC: Für schnelle Maus / VirenKill.TOS / PExe-Mon.ACC: Zeigt alle ausgeführten Programme mit ihren Parametern an, nützlich für TGS

PD-Disk:197

IQ_Test: Testen sie ihre Intelligenz (s/w) / CZ_Split: Midi-Utilities in C / Speech Synthesizer v2.0: Sprachausgabe auf dem ST / Riesenzahlen: Rechnet mit Zahlen bis zu 30.000 Stellen / Clip-ST Nr.3: Diskettenzeitschrift

PD-Disk:198

Terminalprogramme
Kermit v1.1 / My Term v3.0 / Terminal ST v1.52 / Term-ST v5.02 / Uniterm v2.0a mit Keyboardeditor / VT100: Emuliert ein DEC VT100 Terminal / MRamDisk.Acc: Mike's RamDisk aus USA / Word400.ACC

PD-Disk:199

ACC-Disk
Mini Dos / MM Copy / Breakout / Code Tabelle / Control / Digi Clock / Disk PRT / Disk Pic / Druck / Drucken / Emulator / Free Ram / Free-Ram / Hex Calc / Life Game / MAD / Minos / Minos Fin / MM Copy / M Ram Disk / Nec Treiber / Notizblock / Puzzle / Puzzle 2 / RAM D97K / Ram Frei / Reversi / Riteman / R Backup / Shell / Spool / Termin / Termprot / Uhrzeit / Word 400

PD-Disk:200

Kopier-Disk
Bitte ein Bit!: Kopierprogramm mit graf. Trackanzeige v2.0 und v2.4 / Mega-Copy: v 1.8 Umfangreiches Kopierprogramm mit Musik / E_Format: v1.10 Kopier- und Formatierprogramm / Diskformatter für 410 bis 820 kb pro Disk / F_Copy v2.0: Kopierprogramm mit automatischer Formaterkennung / Form 915, Copy 915: v1.0 Formatiert und kopiert 915 kb Disketten / Format: Istari Superformat v1.0 / Hyperformat: v2.56+: Formatiert bis 929 kb pro Disk / Recover: v1.01 rettet gelöschte Files / SuperFormat v1.4: Formatierungsprogramm

PD-Disk:201

Spiele-Disk
Diamond Miner: Ein Spiel ähnlich Boulder Dash, schnelle Grafik und 30 Level, eingebauter Editor (s/w) / Galaxy Sprengmeister: Strategisches *Brettspiel* (s/w) / Mac Pan: Schnelle PacMan Version (s/w) / Hack: Einfaches Dungeon & Dragons Spiel ohne Grafik / Monopoly: Monopoly-Spiel

PD-Disk:202

Spiele-Disk
Bolo: Demoversion, nur 1. Level / Das Horror Schloß: Abenteuerspiel (s/w) / Vector Invasion: Tolle Vectorgrafiken(s/w) / Inversi: Reversi (s/w) / Maziacs: Labyrinthspiel in Omikron-Basic (s/w) / Patience: Franz. Kartenspiel v1.1 (s/w)

PD-Disk:203

Drucker-Disk
FX85, HR15 Druckertreiber für 1st_Proportional / Datobert: Graphdat v1.1, grafische Auswertung von Daten über Kuchen-, Balken und 3D-Diagramme / Disketten Etiketten Drucker: Viele Bilder für die Etiketten (s/w) / Etiketten Druckprogramm: Update von PD 165, nun 47K lang / Expert: Label Expert v3.0, Etiketten für Video, Audio,

5.25", 3.5", Ordner, usw. / Hardcopy: Druckt Hardcopy verzerrungsfrei aus / NEC Manager + P6Code: Drucker konfigurieren mit ACC / Nec Emulator: Konvertiert 9- in 24-Nadeldruckertreiber / VarSpool: Druckerspooler / STX80: Druckt Monochrombilder auf STX80 Drucker / UniDruck: v1.10 Druckprogramm speziell für Listings mit dtsch. Umlauten, in Pascal mit Source / Druckertreiber Wordplus für: FX1000, SP1000-I, NL10 (5x)

PD-Disk:204
Datenbank-Disk
Adress: Adressenverwaltung von Brinck v2.3, incl. Gfa Sourcecode / DiskSort v0.2: (Katalog.Prg) Diskettenverwaltung unter Gem (s/w) / Konto: Priv. Kontoführung (s/w) / K_Kasten: Kartei-Kasten v2.10 von G.Konrad / PrgVerwaltung: v1.0 Diskettenverwaltung ohne TOS / Transist: Transistordaten vergleichen / VLFibu: Finanzbuchhaltung in Deutsch aus Frankreich, ohne Gem (s/w) / Zeitmanager v1.1

PD-Disk:205
Utility-Disk
AR: Archive Vers. 2.3 / ChangeHz: Bildwiederholfrequenz von Farbmonitoren auf 60 Hz setzen=besseres Bild / Compress: Zu STSQ, STUSQ kompatibler Archiver/ CT Night: Schaltet Screen nach 120 sek dunkel, wenn keine Eingaben erfolgten / Setup: Ramdisk für dbMAN / RamXXX: 3 Ramdisks incl. Sourcecode von Focus GmbH / DMon1_4: Disk-Monitor v1.4 / SME: Simple Mouse Editor-eigenes Maussymbol erstellen / Helfer: Diskeditor v1.0 mit Diashow (s/w) / Joshua: Diskeditor / Disk_Utl v1.0: liefert Infos über Diskettenstruktur, incl. Pascal-Source / Edimax: Texteditor, s/w und Farbe / Encrypt: Verschlüsseln von Daten, v1.0 / Gfa-Utl: 26 kleine Basicprogramme / Install: Installiert Harddisktreiber auf Festplatte, booten Sie nun direkt von Harddisk / MemTest: Testen der Speicherbausteine / Diskname: Diskettennamen vergeben / NoFlash: stoppt Cursorblinken / Ramdisk: aus CP / Ram!xxx: 10 Ramdisks von Articulate Software / ToRam: Kopiert Files automatisch in Ramdisk / SectEdit: Diskeditor v1.01, nur Farbe / TLGE: The Little Golem Editor v0.3 (Diskeditor) / Retten: Zerlegt lange Files / Uhr: Zeit und Datum stellen, Zeit wird in Menuleiste angezeigt / Verify: Schaltet verify an und aus

PD-Disk:206
AbinsRam v2.2: Kopieren von Files und Ordnern in Ramdisk / CLI: command line interpreter mit batch jobs v0.19 / Diskmon: Diskeditor unter Gem aus der Schweiz (s/w) / FSelect: Neue File-Select-Box / GfaBasRO: Gfa-Basic Run-Only-Interpreter / Turbodos v1.5: Beschleunigt Disk I/O / IBMDrive: Formatiert und liest IBM-Format / MaxiDisk: Ramdisk komprimiert Inhalt bis 4 Mega resetfest / NewCrypt: Kodieren von Files / PC_Ass: Assembler für Sharp Pocket Computer / RamDisk: Resetfeste Ramdisk aus c't v1.5 / Run Gems: Lädt Gem-Programme automatisch aus Ordner 'AUTOGEM' / SmallMon: v2.0 Diskmonitor (s/w) / View: Anschauen von Textfiles vom Desktop aus mit vielen Optionen / VDU v2.0: Viruskiller aus Holland

PD-Disk:207
Digitalisierter Sound von Pet Shop Boys " always on my mind " mit Gesang, Länge: endlos

PD-Disk:208
Layout ST Demoversion: Platinenlayouts erstellen, Test in ST VISION Nr. 7 / EasyText: Hervorragendes Textverarbeitungssystem mit Hilfefunktionen, eingebautem Rechner, Blockfunktionen, Einrücken, Formatieren, Fettschrift, Unterstreichen. / Checksummer v2.1: Nummeriert Sourcecode

PD-Disk:209
Grafik-Disk
Kubu Draw: Pixelorientiertes Zeichenprogramm mit allen wichtigen Funktionen, nur s/w / Studio:Technisches Grafikprogramm bis DIN A4 und Bemaßung, nur s/w

PD-Disk:210
Sprach-Disk
AdvSys: Textadventure Construction Set, incl. Demoadventure: Columbus / F Shell Vers. 1.11: Oberfläche für ProFortran 77 / Pascal Utility: Oberfläche für ST Pascal+ PasHelp: Syntaxcheck für Pascal / TBI 68K: Gordo's MC 68000 Tiny Basic, Vers. 2.0 / Omikron Basic: Run-Only Interpreter + Beipielprogramme

PD-Disk:211
Sprach-Disk
XLISP: Vers. 1.5b / Tempelmon 1.8: Maschinensprachemonitor und Debugger / Volks-Forth 83: Vers. 3.7, ein Update zu PD 23-24 / Assembler: A68 Vers. 1.0 / Super Copy 520 ST+: Kopierprogramm mit graf. Trackanzeige / Lattice C Utilities:
XD: hexadezimaler Dump, PR: Textausgabe mit Zeilennummern,
NM: Ausgabe einer Symboltabelle, AR: Archiver,
STRIP: entfernt Symboltabelle aus Programm

PD-Disk:212
Demo-Disk
Lachsack: Digitalisiertes Lachen - urkomisch!
10 s/w (PI3) Bilder: Cougar, Death, Helpers, Pacpic, Portait, Rnasance, Rowhouse, Seascape, Stoneage, Tension

PD-Disk:213
Musik-Disk
Midi Utility / Pop Minis (Bad in - Michelle - Popcorn - Ruby) / Synthi / High End 2.0 / Sprachgenerator / Sound STQ (19 verschiedene Klänge)

PD-Disk:214
Musik-Disk
Digi-Soundgenerator für drei verschiedene Längen mit zwei Demosounds.

PD-Disk:215
----**** Alf Special ****----
Super-Sonder-Disk 1
21 Alf Bilder (PI1) & Show Pic 2

PD-Disk:216
----**** Alf Special ****----
Super-Sonder-Disk 2
11 Alf Bilder (PI1) & Show Pic 2

PD-Disk:217
Demoversionen von TommySoft Produkten:
Mega Paint: Ein pixelorientiertes Zeichenprogramm
Star Trash: Das bekannte Weltraumspiel

Bis zur nächsten Ausgabe:
Euer Matthias

PD-TOPHITS

Ausführliche Erklärungen zu Spitzenprogrammen aus unserer PD-Library

Fußballmanager
von PD-Disk 184

Wer wollte nicht auch schon einmal einen Fußballverein managen!
Mit dem Programm 'Fußballmanager' ist dies nun möglich. Nach dem Spielstart tauch man in die faszinierende Welt des Fußballgeschäfts ein. Übernimmt den Verein seiner Wahl und muß nun versuchen ihn von der 3. Liga aus aufzubauen, um schließlich in die 1. Liga auf zu steigen und Meister zu werden.

Auf diesem Weg stößt man natürlich auf all die 'kleinen' Schwierigkeiten mit denen ein Fußballmanager zu kämpfen hat. So muß man Spieler an- bzw. verkaufen, die Manschaft aufstellen, die Eintrittspreise festlegen, das Stadion vergrößern bzw. -kleinern und vieles mehr.

Am Anfang wird erst einmal die Anzahl der Spieler eingestellt (max. 3 Stück). Diese geben dann ihren Namen neben 'ihrem' Bild ein. Zu guter letzt wird noch die Spielstufe eingestellt. Diese reicht auch von eins bis drei, wobei die Stufe eins völlig Ausreicht.

Jetzt entlich geht mit dem ersten Spieltag los. Mit einem Startkapital von 2.000.000,00 DM muß man sich nun durch die dritte Liga kämpfen.
Zuerst wird erst einmal eine Manschaft aufgestellt und versucht dabei den Defensiv- und Sturmteil auf eine möglichst hohe Bewertung zu bekommen. Dann schaut man sich den nächsten Gegner an und wird - wenigstens am Anfang - erschreckt feststellen, daß seine Bewertung im Defensiv- und Sturmteil um einiges höher ist.

Will man ein wenig Geld ausgeben kauft man sich am Besten einen Spieler. Diese sind in Defensiv- und Sturmspieler aufgespaltet und Kosten so je nach Spielstärke zwischen 250.000,00 DM und 1.100.000,00 DM. Oder will man vieleicht sein Stadion ausbauen? Dies ist aber am Anfang nicht sehr Sinnvoll, da höchstens zu Pokalspielen dieses Ausverkauft ist.
Reicht einem dabei das Geld nicht mehr, so nimmt man eben einen Kredit auf. Dies geht bis zu einer Summe von 200.000,00 DM.

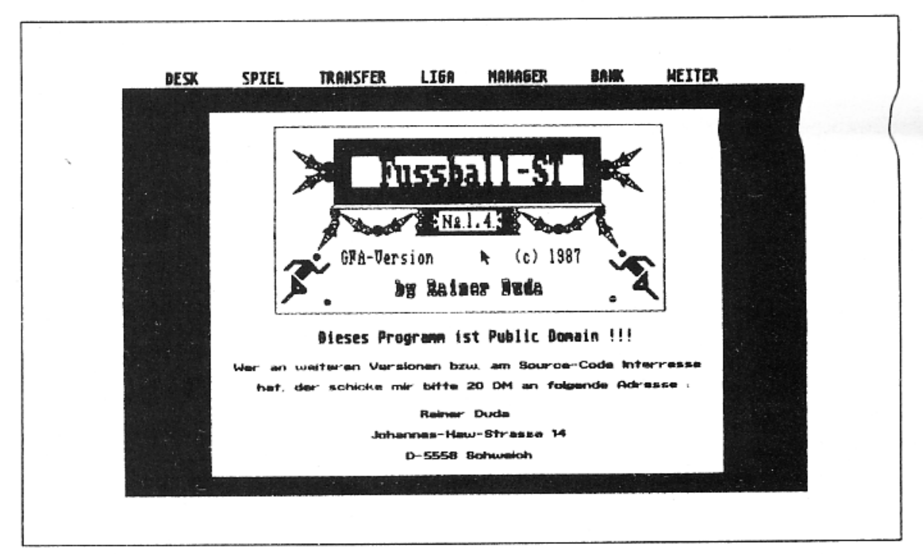


BILD 1: Fußballmanager

Aber jetzt läßt man erst mal den nächsten Spieler seine Einstellungen vornehmen. Haben alle ihre Einstellungen vorgenommen, so findet auch gleich das erste Spiel vor einem auf dem Bildschirm statt. Es wird in neun minütigen Abständen der Spielstand angezeigt und zusätzlich eine Bewertung, die jenachden ob man in Führung liegt oder im Rückstand ist ein lustiges Symbol anzeigt.

Ist das Spiel vorbei bekommt man die Einnahmen und Ausgaben des Spieles angezeigt. Dies währen z.B. die Einnahme der Eintrittskarten und die Ausgabe an Spielergehälter, Reisekosten etc..
Unter dem Strich bleibt dann ein gewisses Sümmchen welches jenachdem ob es positiv oder negativ ist zum eigenen Geld dazu addiert oder davon abgezogen wird.
Als nächstes findet nun das Spiel des nächsten Spielers statt. Sind alle Spiele gespielt und Bewertungen vorgenommen, kommt man in den zweiten Spieltag wo man wieder alle Einstellungen vornimmt. Aber man sollte auf jeden Fall die Manschaft neu Aufstellen. Da es in jedem Spiel Verletzungen und manchmal auch rote Karten gibt.

Ist das Spiel weiter vortgeschritten, so wird es interesant sich den Spielverlauf in Tabellen und Grafiken anzusehen. Dort sieht man den eigenen Auf- bzw. Fortstieg.

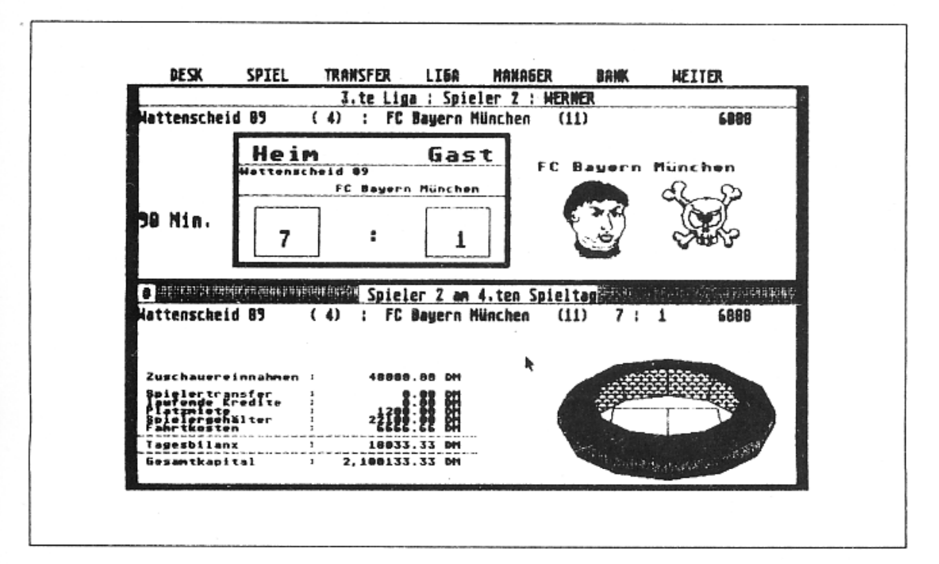


BILD 2: Fußballmanager

Abschließend kann man sagen, daß der Spielablauf ist sehr ansprechent und mit vielen Grafiken untermalt ist.

Also wer wollte nicht schon einmal der Topmanager sein? Hier lernt ihr es.

Fuffy

von PD-Disk 31

Mit Fuffy ist eine schöne Version des Kniffelspieles auf dem Atari ST entstanden. Das Programm ist voll GEM gesteuert und somit einfach zu bedienen.

Mit bis zu vier Personen dieses super Würfelspiel auf dem Computer austragen. Haben alle Spieler ihre Namen eingegeben, so geht es auh schon gleich los.

Der Bildschirm ist in vier Blöcke eingeteilt:

1. Menueleiste
2. Punkteliste
3. Spielerliste
4. Aktionsteil

Wobei der Aktionsteil wiederum in drei Blöcke eingeteilt ist:

1. Würfelbecher mit 4 Aktionspfeilen.
2. 5 Würfel (bei Spielstart nicht sichtbar).
3. Wurfanzeige mit der Möglichkeit die Würfel zu Ordnen.

Zu Beginn wird der erste Spieler in der Wurfanzeige zum würfeln aufgefodert. Dazu muß im Aktionsteil der Aktionspfeil 'würfeln' durch Mausklick betätigt werden. Jetzt erscheinen die 5 Würfel des ersten Wurfs in unsortierter Reihenfolge. In der Wurfanzeig besteht die Möglichkeit die Würfel zu ordnen.

PUNKTE-LISTE		ThoPe	KaWa	Ma...	
1er	nur Einser	2	2	3	1
2er	nur Zweier	6	6	6	8
3er	nur Dreier	9	9	9	3
4er	nur Vierer	12	0	12	12
5er	nur Fünfer	10	15	10	15
6er	nur Sechser	18	18	24	18
OBERER TEIL	SUMME ◊	57	50	64	57
BONUS AB 63	35 Punkte	0	0	35	0
DREIERPASCH	alle Augen	23	11	17	22
VIERERPASCH	alle Augen	26	27	0	26
FULL-HOUSE	25 Punkte	25	25	25	25
KLEINE STRASSE	30 Punkte	30	30	30	30
GROSSE STRASSE	40 Punkte	40	0	40	40
F U F F Y	50 Punkte	50	0	50	0
C H A N C E	alle Augen	20	23	22	22
UNTERER TEIL	SUMME ◊	214	116	184	165
+OBERER TEIL	SUMME ◊	57	50	99	57
GESAMT -	SUMME ◊	271	166	283	222

ENDE-MELDUNG

Das Spiel ist zu Ende !

Wählen Sie bitte anschließend in der Menü-Leiste den Punkt 'SPIEL' an. Sie haben dann die Auswahl zwischen:

Neues Spiel..
Wenn Sie mit den gleichen Mitspielern ein neues Spiel machen wollen, wählen Sie bitte den Punkt '..NEU'.

Neue(s) Spiel/Spieler..
Wenn Sie ein neues Spiel, aber mit neuen Mitspielern machen wollen, wählen Sie bitte '..ABBRUCH' (Auswahl ◊ 2)

Spiel verlassen..
Wählen Sie bitte den Punkt '..ENDE'

OK

BILD 3: Fuffy

Nach dem ersten und zweiten Wurf kann man einselne Würfel wegnehmen und diese neu würflen oder mit dem Wurf auf eine, für einen selber ausreichende, Augenkombination setzen.
Nach dem dritten Wurf muß mit der Augenkombination allerdings in der Spielerliste gesetzt werden.

Der Spielaufbau der Punkteliste dürfte ja altbekannt sein. In der oberen Hälfte sind die einer bis sechser, bei denen nur die Punkte der Selben zählen. Ist man mit der Gesammtpunktzahl dieser Hälfte über 63 Punkte so erhält man einen Bonus von 35 Punkten.
In der unteren Hälfte ist der Dreier- und Viererpasch. Bei beiden zählen alle Augen der Würfel. Als nächstes kommt das Fullhaus mit 25 Punkten, dann die kleine Straße mit 30 Punkten und die große Straße mit 40 Punkten.
Der 'Fuffy' ist der Höhepunkt und wird nur schwer ereicht, bei ihm muß man fünf mal die selbe Zahl würfeln und dies bringt einem 50 Punkte. Zu guter letzt noch die Change, in sie kann man eine beliebige Würfelzusammenstellung eintragen. Es zählen alle Punkte der Würfel. Ganz unten steht noch das Gesammtergebnis.

In den Menuleisten ist eine Abfrage der Würfelstatistik - welche Würfel wie oft gewürfelt wurden - und eine Spielerstatistik - Gesammtanzeige aller Spiele und Anzeige aller einselnen Spiele - grafisch und tabellarisch implementiert.
Zusätzlich sind viele Hilfstexte vorhanden, die einem aus jeder Situation einen Ausweg beschreiben. Diese die jeweils obersten Punkte in jeder Menu-

leiste. Und beschreiben die Menuleiste in der sie stehen. Somit kann man dieses Spiel als absolut Anwenderfreundlich bezeichnen.

Zum spielen mit diesem Programm wünsche ich euch jetzt schon viel Spaß, denn es ist toll aufgemacht.

Hypervoc V1.0

von PD-Disk 38

Hypervok ist ein Programm zum lernen und über von Vokabeln einer beliebigen Fremdsprache. Es ist voll GEM-Gesteuert und somit leicht zu bedienen.

Im Programm stehen 10 Arbeitsspeicher zur Verfügung, welche mit maximal 199 Vokabeln pro Datei belegt werden können.
Die einselnen Arbeitsspeicher können mit verschiedenen Dateien belegt werden, welche dann auch wieder abgespeichert werden können. Zusätzlich besteht die Möglichkeit Dateien von der Diskette zu löschen (scratchen).

Das wichtigste ist sicherlich die Möglichkeit neue Listen zu erstellen. Diese werden relativ einfach eingegeben, und zwar in der Form:
Fremdwort,
deutsche Bedeutung <RETRUN>.

Dazu gibt es noch eine Reihe von Hilfsfunktionen:
Da währen zum einen die Funkion 'stop,' mit der man den Eingabemodus beendet und zum anderen die Funktion 'zurück,' mit ihr kehrt das Programm zu der zuletzt eingegebenen Vokabel zurück, welche dann editiert werden kann.
Zusätzlich besteht die Möglichkeit mit dem Menupunkt 'verbessern' eine neu erstellte oder eine bestehende Liste zu korigieren und somit die noch vorhanden Fehler zu beseitigen.

Um zwei Listen zu verbinden ruft man einfach den Menupunkt 'verbinden' auf. Es werden jetzt zwei Listen, welche von einem bestimmt werden in einer dritten zusammengefaßt.
Dies ist gut geeignet um eine Liste zu erweitern oder um mehrere kurze Listen zu einer großen zusammen zu fassen. Als Resultat entsteht also eine neue dritte Datei in der die beiden alten zusammengefaßt wurden.
Aber, und das ist Wichtig, die beiden alten Dateien bleiben bestehen.

Aber wir wollen ja nicht nur die Vokabeln eingeben, nein das Lernen der Fremdwörter ist ja das eigentliche Ziel des Programms. Um dies zu ereichen, gibt es erst einmal den Menupunkt 'lernen'. Hat man diesen Selektiert, so muß ein Bereich von max. 20 Vokabeln angegeben werden, in dem gelernt werden soll.
Der Computer schreibt die Fremdwörter nun auf die rechte Seite und fragt sie von oben nach unten ab. Beim Abfragen besteht auch noch die Möglichkeit nach Eingabe der Übersetzung, sich die richtige Lösung anzeigen zu lassen.
Hat man alle Bereiche der Liste auf diese Weise geübt und ist sich sicher die Vokabeln zu beherschen, wagt man sich zum Menupunkt 'testen' vor.

Der Menupunkt 'testen' diehnt zum (wie sollte es auch anders sein) testen der mit lernen geübten Vokabeln. Zuerst wird abgefragt wie oft der Computer fragen soll. Als Vorgabe ist er auf den Wert 90 gesetzt.
Er greift nun zufällig eine Vokabel aus der Liste und wartet auf ihre Antwort. Es wird ihnen dann mitgeteilt, ob ihre Antwort richtig oder falsch war. Richtig beantwortete Vokabeln werden nicht mehr abgefragt.
Hat man alle Fragen beantwortet, so wird sogar eine Benotung ihrer Leistungen vorgenommen. Dies geschieht im Punktsystem der gymnasialen Oberstufe (von 1 bis 15).

Zu guter letzt kann man sich unter dem Menupunkt 'Bildschirm' alle Vokabeln einer *Liste* übersichtlich ansehen. Und mit dem Menupunkt 'Drucker' schließlich auf dem Drucker ausgeben. Der Bereich der auszudruckenden Vokabeln ist beliebig einstellbar. Dazu können auch noch die Schriftgrößen und -breiten eingestellt werden.

Was braucht man also mehr zum Vokabellernen als dieses Programm!?!

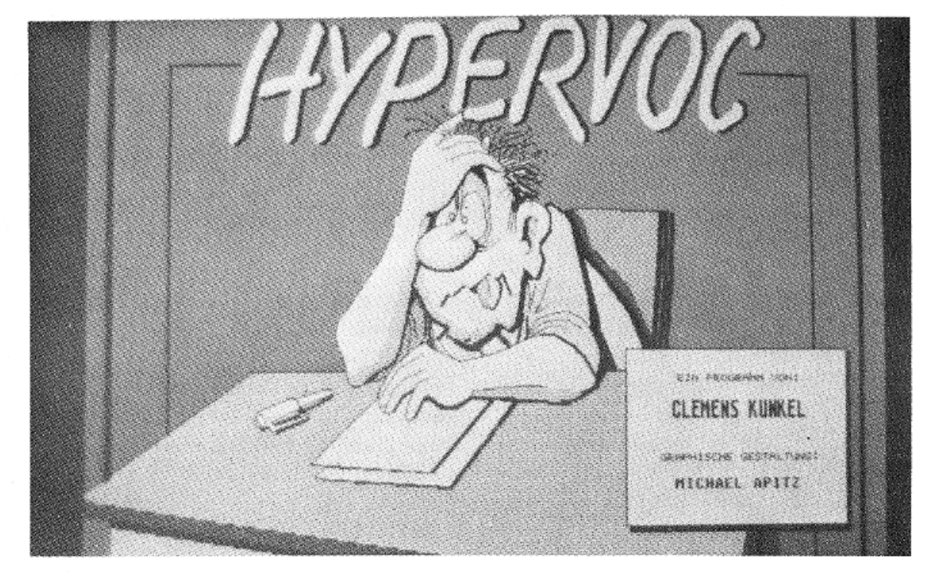


BILD 4: Hypervoc V 1.0

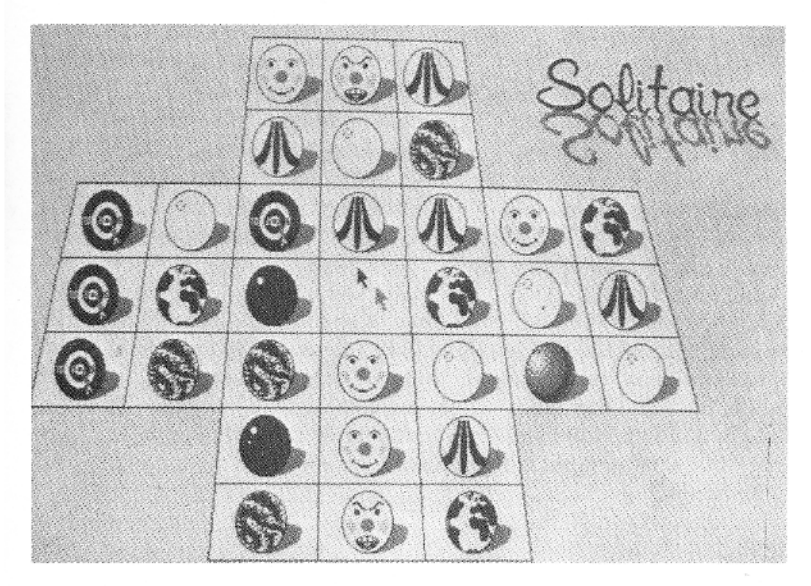


BILD 5: Solitaire

Solitaire
von PD-Disk 96

Dieses Spiel ist eine sehr schöne Variante seiner Art.

Sinn, Ziel und Zweck dieses Spiels ist es, alle Kugeln die auf einem 3D-Spielbrett liegen, durch überspringen, vom Brett zu nehmen. Bei dieser Variante kann man zusätzlich eine Vielzahl von Einstellungen vornehmen.

Aber zuerst noch einmal zu der Spielerklärung:
Das Spielbrett ist in gleich große Quatrate eingeteilt. Auf jedem dieser Quatrate liegt am Anfang eine Kugel. Ein Feld ist nicht belegt, dieses nennt man auch das Start-, Endfeld.
Man muß jetzt, wie oben schon gesagt, versuchen durch überspringen der Kugeln alle vom Brett zu nehmen.

Dies geschieht unter den folgenden Regeln:
1. Es darf immer nur eine Kugel übersprungen werden.
2. Es darf nur waagrecht und senkrecht gesprungen werden.
3. Es darf zwischen der zu überspringenden Kugel kein Freiraum sein.
4. Die letzte Kugel muß an der Stelle liegen, an der am Anfang der Freiraum war (ideales Ende).

In dieser Variante kann man zusätzlich noch die Farbe der Kugelneinstellen. Dies geht von schwarz über weiß und mamoriert bis zu Kugeln mit Gesichtern, Weltgloben, Zielscheiben, etc..
Auch besteht die Möglichkeit, zwischen planen und 3D Kugeln zu wechseln. Und es sind außerdem die englische und französiche Variante implementiert.

Zur Verdeutlichung:

englische Variante	französiche Variante
+++	0++
+++	+++++
+++++++	+++++++
+++0+++	+++++++
+++++++	+++++++
+++	+++++
+++	+++

Wobei die Null das Start-, Endfeld ist.
Als Zusatz gibt es einen Demomodus in dem die Lösung der englische Variante gezeigt wird. Damit man weis, daß es wirklich eine Lösung gibt.

Für Leser die das Spiel kennen, wird dies zwar alles 'alter Kaffee' sein. Aber das Spiel ist trotz dem auch diesen Lesern zu empfehlen, allein schon wegen seiner schönen ideenreichen Aufmachung.

Bolo
von PD-Disk 202

Bolo-PD das Spiel der Spiele, kennen sie Akanoid dann müssen sie Bolo kennen lernen. Dieses Spiel ist so gut, daß man direkt traurig ist, wenn die erste Stufe vorbei ist. Denn in Bolo-PD kann man nur die erste Stufe spielen. Aber was soll's, bei diesem Spiel ist man schon froh die erste Stufe spielen zu können.

Bolo stammt von Application Systems Heidelberg und dies sagt schon genug aus (STAD - SIGNUM - FLEXDISK - etc.).

Aber nun zum Spiel:
Bolo ist eine Breakout-Variante d.h., daß der Spieler einen Schläger steuert der auf dem Bildschirm in 'alle' Richtungen bewegt werden kann.
Links, oben und rechts sind Wände unten ist nichts. Gelangt nun der Ball an den unteren Bildschirmrand ins Nichts, so ist dieser verloren.

Der Sinn des Spiels ist es nun nicht nur den Ball im Spiel zu halten, nein man muß auch die Steine die sich im oberen Teil des Spielfeldes befinden, durch berühren mit dem Ball reduzieren bis keine mehr da sind. Dann gelangt man in die nächste Spielstufe.

Die Steine haben unterschiedliche Auswirkungen auf den Ball. So wird der Ball von den hellen Steinen, je nach dem aufschlags Winkel, wieder in einem Winkel abgelenkt.
Die schwarzen Steine sind so eine Art von Teleportern. Trifft man diesen Stein an einem bestimmten Punkt, so das er verschwindet, kommt der Ball an einem anderen wieder auf dem Bildschirm heraus.

Berührt der Ball in der zweiten Spielstufe einen schwarzen Stein, so wird der Ball so langsam, daß das Spiel unspielbar wird.

Zarge
von PD-Disk 188

Zarge ist ein Puzzlespiel ganz besonderer Art. Dieses Programm macht aus jedem Monochrombild im Screenformat (32000Byte) ein Puzzle.
Dabei wird das Bild in gleichmäßige Quatrate zerteilt, gemischt und wieder auf dem Bildschirm gebracht. Nur ein Feld ist frei.
Um nun das Bild wieder in seine ursprüngliche Form zu bringen, muß man die Bildstücke auf dem Bildschirm waagrecht und stenkrecht hin und her schieben.

Dieses Programm ist einfach gehalten, aber durch seine nahezu unendlichen Schwierigkeitsstufen und Variationsmöglichkeiten (die nur vom eigenen Vorrat an Bildern abhängt -> siehe auch unsere Bilderdisketten) immer wieder interesant und faszinierend.

Ich persönlich spiele dieses Spiel gerne und oft um micht zu entspannen.

StarTrash
von PD-Disk 217

StarTrash auf PD-Disketten, das gibts doch nicht !?!

In der Ausgabe 5 wurde es beim Preisrätsel noch als Hauptpreis verlost und jetzt in der PD-Ecke? Des rätsels Lösung ist, daß es
jetzt eine Demo-Version von Star Trash gibt. In dieser kann man, man höre und staune die erste Stufe komplett wie in der richtigen Version spielen. Und jetzt fragt einmal unsere Hauptgewinner, welche von ihnen jemals die erste Stufe geschaft haben!

Doch was ist eigentlich Star Trash?

Dazu die Einleitung:
Der Weltraum - unendliche Warteschlangen. Wir stempeln das Jahr 2215. An Bord der Behörden befinden sich so berühmte Persönlichkeiten wie Captain Flirt, Mr. Spoil, Dr. Jeckel und Mr. Scotch.
Sie sind auf der Suche nach den verloren gegangenen Aktenordnern der Galaxis, die im Laufe einer Bordparty versehentlich auf Star Trash VII gebeamt wurden.
Unangenehm bei der Suche nach den verschollenen Akten wirkt sich die Tatsache aus, daß sich auf Star Trash VII die einzige konföderierte Besserungsanstalt der Galaxis befindet.
Im 20. Jahrhundert nannte man diese Institution noch "geschlossene Heilanstalt für psychisch Erkrankte". Zum Glück wurden diese barbarischen Methoden längst abgeschafft.
Das Ergebnis befindet sich auf Star Trash VII !

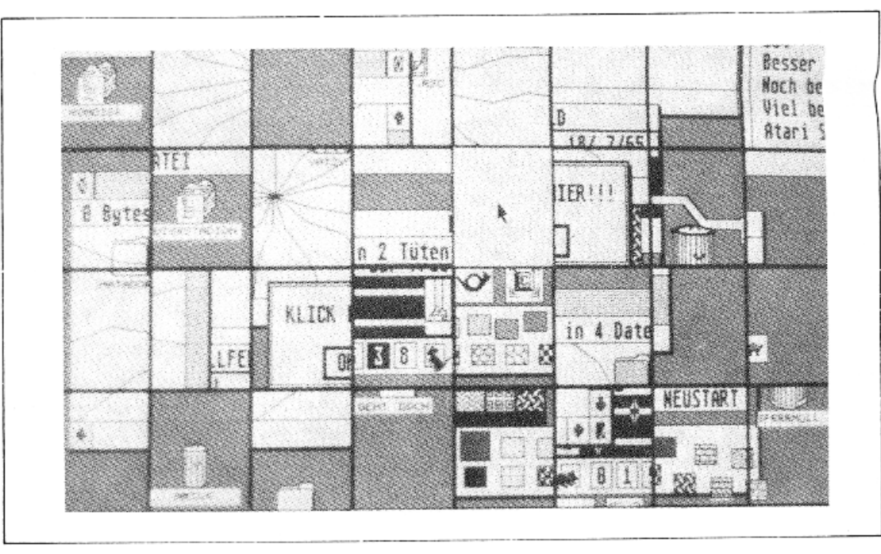


BILD 6: Zarge

Auf der Suche nach den Akten muß der Neuronenflummi eine vielzahl von Abenteuern bestehen. Und auch den den Irren auf dem Planeten nicht zu nahe kommen.
Dies ist nicht leicht, da sich im laufe der Zeit ein wahres Batallion an Irren dort angesammelt hat. So gibt es z.B. Rollende Mülltonnen vom Planeten Saubermann, häßliche Spinnen aus dem Spiralnebel Tarantula und viele andere irrwitzige Gestalten, die von einem nieh das Beste wollen.

Doch was ist denn eigentlich ein Neuronenflummi ?
Dieses kleine Gerät ist ein hüpfender fergesteuerter Roboter, mit dessen Hilfe der Spieler versucht, sich seinen Weg durch Star Trash VII zu bahnen.

Und wenn euch diese Vorgeschichte noch nicht reicht um die PD-Disk zu holen, dann lest doch noch einmal in unserer Ausgabe 5 ab Seite 38 den ausführlichen Bericht über Star Trash.

Sortierte Disketten
von PD-Disk 198 bis 216

Neu in unserem Sortiment sind die sortierten Disketten. Diese sind nach Kriterien wie Drucker-Disk, Spiele-Disk, Utility-Disk, Asseccory-Disk, etc. sortiert. Das bringt den absoluten Vorteil, daß man mit einer Diskette ein ganzes Gebiet abdeckt und somit nicht mehrere Disketten kaufen muß, bis man das richtige Programm für einen selbst gefunden hat.
Braucht man z.B. ein gutes Terminalprogramm, so kauft man sich die PD-Disk 198 DFÜ.
So liegt einem eine Diskette mit den verschiedensten Programmen dieser Art vor. Damit kann man sich einfach das für einen selbst am Besten geeignete Programm heraus suchen.

Easy-Text

von PD-Disk 208

Easy-Text, ein Programm von sehr guter Qualität 'flatterte' uns vor wenigen Tagen ins Haus.

Dies ist ein Textverarbeitungsprogramm, welches für den Heimbenutzer völlig ausreichend ausgestattet ist - und somit den Kauf von solchen Programmen überflüssig macht.

Dieses Programm beinhaltet nicht nur alle wichtigen Textverarbeitungsfunktionen, nein, es bietet sogar noch mehr!

Als Standardfunktionen kann man z.B. Laden, Speichern, Unterstreichen, Fettdruck, etc. bezeichnen. Doch was bietet es sonst noch an Zusätzen?

Zum einen kann man sich die jeweilige Position des Cursors anzeigen lassen, welches speziell bei längeren Texten zur Übersichtlichkeit dient. Als nächstes steht eine zwar einfache, aber dennoch hifreiche Funktion zur Verfügung, ein Taschenrechner.

Die Blockoperationen sind auch voll implementiert. So kann man einen beliebigen Start- und Endpunkt, auch über mehrere Seiten hinweg, definieren. Der Block kann verschoben, kopiert, gespeichert, geladen und gelöscht werden. Damit haben wir alle notwendigen und hilfreichen Blockoperationen versammelt. Nicht vergessen sollte man den finde/ersetze Modus. In ihm kann man schnell und einfach Zeichenketten austauschen. Die Seitenlänge ist natürlich ebenso einstellbar wie die Zeilenbreite.

Wobei wir auch bei dem sogenannten Formatieren wären. Hat man das 'Lineal', also die Zeilenbreite, verändert, ist es oft wünschenswert, den schon geschriebenen Text dem neuen Lineal anzupassen. Dies geschieht mit der Funktion 'Neuformatieren'. Mit ihr werden sämtliche Zeilen nur mit soviel Wörtern gefüllt, daß keines über das Lineal hinausschaut. Dies geschieht bis zum nächsten Absatz - was von Vorteil ist, wenn man z.B. Absätze in verschiedenen Breiten schreiben will. Das Beschriebene nennt man auch Flatterrand.

Aber es ist auch noch der sogenannte Blocksatz implementiert. Bei ihm werden nicht nur so viele Worte in jede Zeile geschrieben, wie das Lineal lang ist, hier werden zusätzlich die Worte so verschoben, daß der Text links und rechts immer die gleiche Breite hat. Dies sieht nicht nur professioneller aus, es dient auch der Übersichtlichkeit und dem besseren Aussehen.

Die Möglichkeit den Bildschirm zu invertieren, d.h. daß alles weiße schwarz wird und umgekehrt, sollte man nicht vergessen. Dies ist eine Wohltat für die Augen. Wo wir auch schon bei der Lauffähigkeit des Programms wären. Es läuft leider nur in Monochrom, so daß alle 'Atarianer', die keinen Monochrommonitor besitzen, leider dieses Programm nicht benutzen können.

Ansonsten ist noch zu sagen, daß EasyText eine Textlänge von 7950 Zeilen (1040STF) zuläßt. Dies entspricht einer Länge von 120 Seiten (Seitenlänge = 66 Zeilen). Das dürfte nicht nur für den Heimgebrauch ausreichend sein!

Was man aber auf gar keinen Fall vergessen sollte und was ich sehr begrüße ist, daß dieses Programm eine Frau, C. von Keitz, geschrieben hat. Ich hoffe, daß uns noch mehr solcher Programme in Verwunderung versetzen werden. Und habt keine Angst um uns. Wir werden mit der Brief- und Programmflut schon fertig!

Zum Schluß noch ein kleiner Zeitvergleichstest zu 1st Word Plus ((c)GST-Software), welches man wohl als das Standardprogramm bezeichnen kann:

Laden eines 23 seitigen (1439 Zeilen, 6621 Worte) Textes:
Easy Text 1'03",80
1st Word 0'28",23

Zeilenweises Durchblättern des Textes vom Anfang zum Ende:
Easy Text 1'37",69
1st Word 5'47",44

Die jetzt folgenden Blockoperationen beziehen sich auf einen 969 Zeilen langen Text.

Block definieren:
Easy Text <1"
1st Word 9",20

Block kopieren:
Easy Text 1",50
1st Word 33",57

Block löschen:
Easy Text 1",38
1st Word 33",83

mas

DAS MIDI- UND SOUND-BUCH ZUM ATARI ST

Fertige Software für musikalische ST-Benutzer gibt es schon länger, wer jedoch selbst die Fähigkeiten seines ST auf diesem Gebiet nutzen wollte, sah sich bisher recht allein gelassen. Das mir vorliegende Buch zeigt sich als geeignet, hier etwas Licht in das Dunkel zu bringen.

Der Inhalt wird zur besseren Veranschaulichung von vielen Beispielen in C und/oder GfA-BASIC, Hardcopies und Abbildungen begleitet. Dem Buch liegt eine Diskette mit dem Sourcecode der größeren Beispielprogramme und -Programmteile bei.

Der Geschichte der Musikelektronik ist das erste Kapitel gewidmet. Tatsächlich ist die Nutzung elektronischer Hilfsmittel erst in jüngster Zeit, nicht zuletzt durch die Möglichkeiten des ATARI ST, auch für den (finanziell) minderbemittelten Musiker erschwinglich geworden. Auf die Programmierung des zugegebenermaßen ärmlichen Soundchip des ATARI STs wird nur relativ kurz eingegangen.

Erfreulicherweise wird ein vergleichsweise niedriger musikalischer Kenntnisstand zum Verständnis dieses Kapitels verlangt. Die Autoren fangen praktisch noch einmal bei "Adam und Eva" an und erklären musikakustische Grundlagen sowie theoretische Anfänge der elektronischen Klangerzeugung. Rechteckschwingung, Oszillator und ADSR-Hüllkurve sind damit dem Leser ein Begriff.

Einige Beispiel-Programme in C und GfA-BASIC zeigen, daß bei geschickter Programmierung doch noch etwas aus den 16 Sound-Registern herauszuholen ist. Der weitaus größte Teil des Buches beschäftigt sich jedoch mit der Programmierung des MIDI-Ports des ATARI STs.

MIDI = Musical Instrument Digital Interface

Auch hier wird wieder mit den geschichtlichen Hintergründen analoger und digitaler Sounderzeugung mit Synthesizern und Samplern begonnen. Der Leser erhält einen Überblick über die MIDI-Peripherie, die MIDI-Hardware und deren Anwendungsmöglichkeiten des Aufbaus, der Schaltung und Verkabelung einzelner MIDI-Geräte wie z.B. Synthi-Keyboard, Expander, Sampler oder Drum-Machine.

Relativ ausführlich wird dann auf die Einzelheiten der MIDI-Sprache eingegangen, die wegen ihrer Komplexität in gut strukturierten Häppchen verabreicht wird. Leider können die Erläuterungen nicht allzusehr in die Tiefe gehen, da die MIDI-Geräte verschiedener Hersteller z.B. beim laden und speichern von Klangdaten jeweils unterschiedliche Datenstrukturen verarbeiten.

Wem das selbständige Programmieren seiner Soundanwendungen dann doch zu mühselig erscheint, für den werden einige der meistverbreiteten professionellen Programme vorgestellt.

In Kapitel sechs schließlich gehen die Autoren nochmal auf die digitale Soundverarbeitung mit Digitizer und Sampler ein und stellen auch hier ein professionelles System vor. Wer daraufhin Appetit bekommen hat selbst mit digitalen Klängen zu experimentieren und das entsprechende Bastlergeschick aufweist, kann sich anhand der Bauanleitung im Anhang seinen Digitizer mit immerhin 6 Bit Auflösung selbst bauen. Die dazugehörige Software findet sich im nachfolgenden Assembler-Listing, bzw. auf der beiliegenden Diskette auch als ausführbares Programm. (Es handelt sich hier um eine optimierte Version des Digitizers aus der "68000er" 9/86).

Im Anhang finden sich auch die Source-Listings der größeren Anwendungsprogramme in BASIC und C. Darunter ein Multifunktionsdemo zur MIDI-Programmierung, ein Editor-Programm zum Einlesen, Ändern und Zurückspeichern der Klangdaten eines Synthis sowie nützliche Routinen zur gleichzeitigen Ansteuerung von MIDI und Soundchip. Den Abschluß bilden einige Tabellen z.B. zur Umwandlung von Zahlensystemen, Noten/Frequenzen/Registerwerte sowie ein ausreichendes Stichwortverzeichnis.

Für den Musikfreund, der in Sachen MIDI-Anwendung praktisch noch in den Kinderschuhen steckt und sich z.B. einen MIDI-fähigen Synthesizer zulegen möchte, kann ich dieses Buch ohne weiteres empfehlen. Geeignet ist es auch als Entscheidungshilfe, ob MIDI-Software selbstgestrickt oder professionelle Software gekauft werden soll.

Das 240-seitige Buch ist im Markt & Technik Verlag erschienen und kann für DM 69,- käuflich erworben werden. ISBN 3-89090-528-5. ws

DER 16BIT-MIKROPROZESSOR 68000

Von Joachim Koch, erschienen im Hüthig Verlag

Auf 218 Seiten wird alles Wissenswerte rund um die CPU 68000 zusammengefasst. Doch spult Joachim Koch hier nicht nur das Pflichtprogramm, respektive die seitenweise Erklärung aller Befehle ab: vielmehr werden viele interessante Informationen über die ganze 68000-CPU-Familie, deren Architektur und programmiertechniken vermittelt.

Datenfelder, Stacks und Queues

Gleich nach der Einleitung folgt Grundsätzliches zur Registerarchitektur des 68000 und deren Anwendung zur Verwaltung von Datenfeldern, Stacks und Queues. Es folgen Erläuterungen zu Befehlsformaten und Adressierungsarten, wobei die zahlreichen Adressierungsmodi des 68000 durch sinnvolle Erläuterungen und Hilfszeichnungen ergänzt werden.

Es folgt die Erklärung der Ausnahmeverarbeitung, also die Verarbeitung von Traps. Da das Buch nicht speziell für den Atari verfasst wurde, findet man hier natürlich nur grundsätzliche Informationen über die Trapverarbeitung des 68000; deren Verwendung auf dem Atari wird nicht besprochen.

Klare Befehlsübersicht

Mit dem fünften und sechsten Kapitel kommt man schließlich zum obligatorischen Pflichtteil, der in jedem guten Buch über den 68000 zu finden ist: auf über 120 Seiten erfolgt die detailierte Beschreibung aller Befehle, wobei jedem Befehl mindestens eine Seite gewidmet ist.

Die Beschreibung umfasst neben der umgangssprachlichen Erklärung des Befehles noch den deutschen und englischen Befehlsnamen, eine Zusammenfassung der möglichen Adressierungsarten und falls nötig die genaue Assemblersyntax.

RTL - Register Transfer Language

Die Operation werden ferner in RTL dargestellt, wobei RTL für Register Transfer Language steht, eine Notation zur Darstellung, welche Register sich wie verändern.

Während diese umfassende Befehlserklärung für den täglichen Gebrauch im Zugriff zu langsam ist, kann in den folgenden Kapiteln noch auf zusammenfassende Tabellen zurückgegriffen werden, in denen der Befehl, eine Kurzbeschreibung und die Beeinflussung des Statusregisters gegeben ist.

Auch sind Tabellen vorhanden, nach denen man berechnen kann, wie lange ein Befehl zur Ausführung braucht, was für Programmierer zeitkritischer Anwendungen (z.B. Graphik) unbedingt erforderlich ist. Auch auf den Prefetchmechanismus der CPU und eventuell daraus resultierende Probleme wird dabei eingegangen.

Beispielprogramme

Abgerundet wird das Buch durch einige Beispielprogramme, bei denen grundsätzliche Programmierpratiken wie einfache Tabellenoperationen und erweiterte Rechenoperationen nähergebracht werden sollen. Diese Programme sind natürlich auch nicht rechnerspezifisch und laufen daher auf jedem 68000-System.

Insgesamt ist das vorliegende Buch "Der 16bit-Mikroprozessor 68000" sowohl für den Einsteiger zum Erlernen der Assemblersprache als auch für den Profi als tägliches Nachschlagewerk sehr geeignet. Besonderes Lob verdienen dabei die besondere Übersichtlichkeit der Tabellen und Darstellungen sowie der logisch strukturierte Aufbau.

pky

Titel:

Joachim Koch
Der 16bit-Mikroprozessor 68000
218 Seiten, DM 38.-
ISBN 3-7785-1556-X

Erschienen bei:

Hüthig Verlag
Im Weiher 10
6900 Heidelberg

TOOLS & ALGORITHMEN

Das große GFA-Programmierhandbuch für GFA 2.0 & 3.0 - von Data Becker

Das Buch Tools & Algorithmen von Data Becker soll als Nachschlagewerk für Standard Algorithmen dienen. Diesem Vorsatz wird das Buch im Großen und Ganzen auch gerecht. Denn Algorithmen gibt es dermaßen viele, daß man sie niemals in einem Buch zusammenfassen könnte. Es wurden die gebräuchlichsten als Standard-Algorithmen zusammengefaßt. Diese und andere werden in diesem Buch beschrieben.

Aber was sind denn eigentlich Algorithmen !?!

Ein Algorithmus ist nach dem Fremdwörter-Duden ein "Rechenvorgang, der nach einem bestimmten Schema abläuft". Speziell auf Standard-Algorithmen gesehen, sind dies Lösungen für oft benötigte Problemstellungen (z.B. Sortieralgorithmus) eines bestimmten Programmteils. Diese Lösungen sind in sogenannten Struktogrammen aufgeführt.

Das Aussehen der Struktogramme gleicht einem Ablaufdiagramm, wobei es aber von oben nach unten abgearbeitet wird. D.h., daß keine Bewegungen nach links, rechts oder sogar oben durchgeführt werden.

Der große Vorteil dieser Struktogramme liegt darin, daß sie unabhängig von einer Programmiersprache geschrieben werden. So kann man ein Struktogramm, ist es einmal geschrieben, in jede Programmiersprache ohne Probleme übersetzten.

Aber nun zurück zu den Algorithmen. Diese sind also eine optimierte Lösung zu bestimmten Problemgebieten.

Was bietet das Buch ?

Der zweite Teil, der sich mit Algorithmen beschäftigt und über 300 Seiten stark ist, ist in acht große Teile eingeteilt, die jeder für sich ein Teilgebiet der möglichen Problemstellungen abdecken.

Das erste Gebiet behandelt die Datenstrukturen. Darunter fallen die diversen Sortieralgorithmen, Suchalgorithmen, Ausgabe, Eingabe, Binärbäume, Textverarbeitung mit Blocksatz, Silbentrennung, WordWrap.

Im zweiten Teilgebiet dreht sich alles um die Grafik. Es geht dabei um Vektorräume, Koordinatensysteme, diverse Manipulationen von Grafikobjekten, Kreise, Elipsen, Flächenbestimmung von regelmäßigen n-Ecken - Beliebigen Polygonzügen - etc. und sogar EAN-Barcodes und Metafiles.

Das dritte Teilgebiet beschäftigt sich mit der Finanzmathematik. Dort werden Algorithmen für die Zinsrechnung, Tilgungspläne und Investitionsrechnung vorgestellt.

Das vierte Teilgebiet hat sich die Lösung von numerischen Algorithmen vorgenommen. Dieses umfangreiche Gebiet beginnt bei dem Algorithmus von Euklid, des Siebs des Eratosthenes, der Rechenarithmetik, geht über die Matrizenoperationen, die Lösung von quadratischen Gleichungen, Berechnung der Nullstellen, Integral- und Differentialrechnung, Umrechung und Umwandeln von Zahlen (systemen) bis zur Umrechnung von Winkelangaben in Sekunden.

Das fünfte Teilgebiet beschäftigt sich voll und ganz mit der Statistik und Wahrscheinlichkeitsrechnung.

Das Teilgebiet sechs beschreibt Algorithmen rund um den Kalender. Da wären die verschiedensten Berechnungen und Prüfungen von dem Datum, den Wochentagen, von Tagesabständen und Feiertagen.

Die richtige Strategie findet man in dem siebten Teilgebiet. Dort geht es um die Steuerung und Wegsuche in einem Labyrinth und die Routinenoptimierung. Zu guter letzt werden im achten Teilgebiet noch nützliche Algorithmenschnipsel zusammengefaßt.

Im dritten und letzten Teil bietet dieses Buch die im Namen schon versprochenen Tools für den Atari. Tools sind Hilfsgeräte für bestimmte Gebiete. Hier werden fünf Gebiete behandelt.

Als erstes geht es rund um den Massenspeicher. Dabei werden Suchpfade, Dateiattribute, Dateitypen, Files und ein automatisches Backup-System behandelt. Das zweite Teilgebiet beschäftigt sich mit der Tastaturbelegung. Es werden Tools zur Scan- und Ascii-Codeermittlung und das Anlegen eines eigenene Zeichensatzes erklärt. Als drittes ist der Bildschirm an der Reihe. Behandelt werden dabei Balken- und Tortendiagramme, das Arbeiten mit mehreren

Bildschirmseiten und ein 3D-Funktionsplotter. Im vierten Teilgebiet befaßt sich das Buch mit der Kommunikation. Dabei werden einführede Erklärungen über die Kommunikation mit anderen Rechnern gegeben. Das fünfte Teilgebiet behandelt die Programmiersprachen, mit einem Schwerpunkt auf GFA-Basic. Es wird hier ein REM-Killer und ein Preprozessor vorgestellt.

Die beiden Teile sind natürlich mit Programmbeispielen versehen, was die direkte Anwendung und das sofortige Ausprobieren ermöglicht. Diese Programme sind sämtlich in GFA-Basic geschrieben. Im Teil drei sogar schon mit GFA-Basic 3.0.

Der Anhang besteht aus den folgenden 5 Teilen. Der erste Anhang befaßt sich mit den mathematischen Funktionen. Es wid dort z.B. der sinh(x) aufgeschlüsselt, so daß man ihn selbst programmieren kann. Der zweite Anhang enthält ein Literaturverzeichnis. Dort sind interessante Bücher zu den einzelnen Themen aufgelistet. Der dritte Anhang enthält eine komplette ASCII-Tabelle.

Der interessanteste Anhang dürfte der Vierte sein. In ihm wird auf 30 Seiten die AES-Library von GFA 3.0 erläutert. Dies umfaßt sämtliche Window, Menu, RSC, etc. Aufrufe.

Der letzte Anhang enthält Hinweise für Umsteiger von GFA 2.0 auf GFA 3.0. Es werden dort alle Unterschiede übersichtlich aufgelistet.

Das Buch umfaßt 480 Seiten und eine Diskette, auf der alle Programmbeispiele des Buchs enthalten sind. Die Autoren haben das Buch übersichtlich gestaltet und in einer leicht verständlichen Sprache geschrieben.

Das große GFA-Programmier-Handbuch 'Tools & Algorithmen' wird von Data Becker (ISBN 3-89011-258-7) für DM 59.- angeboten. mas

GFA-BASIC 3.0

Das dazu passende Buch von GFA

Das GFA-Buch 3.0 ist die optimale Ergänzung zum neuen GFA-Basic. Es zeigt die Möglichkeiten und Anwendungsbeispiele in mehreren Kapiteln auf. Durch diese kann man die neuen Befehle erst richtig Ausschöpfen.

Das Buch gliedert sich in folgende Kapitel:

1. Eine Kurzeinleitung in die neue Programmiersprache.
2. Beschreibung der neuen Programmstrukturbefehle.
3. Die neuen Variablentypen und die Befehle zu deren Verwaltung.
4. Die Integerarithmetik, die es in GFA-Basic 3.0 nun wirklich gibt.
5. Die Line-A-Aufrufe und deren Bewandnis.
6. Die neuen Befehle, die die Fehlersuche vereinfachen.
7. Die verbesserten Möglichkeiten der Assemblereinbindung.
8. Das größte und wichtigste Kapitel erklärt den Einsatz der neuen umfangreichen AES-Funktionen.
9. Hier wird GDOS und deren Verwendung in GFA-Basic 3.0 erklärt.

Zusätzlich gibt es noch einen auführlichen Anhang über fast 100 Seiten. Dieser gliedert sich in vier Teile.

1. Die neuen Befehle des Editors inklusive der Kontrol-Komandos.
2. Eine Übersicht über alle neuen Befehle und Änderungen.
3. Die komplette AES-Bibliothek.
4. Die Unterschiede zu GFA Basic 2.xx.

Den Abschluß bilden 17 Seiten Stichwort- und Befehlsverzeichnis.

Am Ende des Buches befindet sich auch noch eine Diskette mit allen beschriebnen Beispielprogrammen. Darunter auch größere Programme, wie z.B. ein Formel-Parser, eine Routine zum Lesen aller Dateinen auf einer Diskette in allen Ordnern, eine Textausgabe in Assembler, ein Debugger, ein Resource-Analyser, ein 3D-Funktionsplotter, Programme zur Dialogboxverwaltung, Pull-Up-Menüs, Fensterverwaltung und vieles mehr.

Dieses Buch gilt als wertvolle Ergänzung zum Handbuch des GFA-Basic's 3.0, da dort 'nur' alle Befehle ausführlich erklärt werden. Denn um eine so komplexe Programmiersprache zu verstehen, muß man nicht nur wissen, was die Befehle bedeuten, man muß sie auch anwenden können.

Das Buch bietet dazu in den jeweiligen Kapiteln Anwendungsbeispiele, die die Befehle in einem interessanten Zusammenhang vorstellen.

Aber was man nicht vergessen sollte, ist die Autoren dieses Buches zu erwähnen. Denn sie haben es mit Kompetenz und dem nötigen Fingerspitzengefühl geschrieben.

Zum einen ist Gottfried P. Engels (25) zu erwähnen. Er ist Tutor für Statistik an der Universität Düsseldorf. Zum anderen Markus C. Görgens (24). Er studiert Mathematik mit Nebenfach Informatik und arbeitet seit längerem mit einem Atari ST.

Das GFA-Basic-Buch zur neuen Version 3.0 wird vertrieben von: GFA Systemtechnik GmbH, Düsseldorf, (ISBN 3-89317-004-9) für DM 59.- vertrieben.

mas

COMPUTERFLOHMARKT

Verkauf, Kaufgesuche, Kontakte zu anderen Mitgliedern

Hier ist sie nun endlich - die Rubrik Kleinanzeigen. Hier kann günstiges Zubehör erstanden werden, oder nicht mehr benötigte Hard- und Software zum Verkauf angeboten werden.

Eine nichtgewerbliche Kleinanzeige kostet DM 5.-. Für ST VISION Mitglieder und Abonnenten ist der Anzeigenservice natürlich kostenlos.

Verkaufe

Atari SF 354 Laufwerk, wenig gelaufen, wegen Umrüstung günstig abzugeben. Preis VS. Thomas Mäurer, Duisburger Str. 296, 4200 Oberhausen 1, Tel. 0208/20883.

Diverse XT und AT Gehäuse aus Lagerauflösung zu verkaufen, Preis zwischen DM 30 - 60.-.
Ca. 50 (!) Meter 8 poliges RS-232 Kabel, Typ 'Clear Signal' der Firma Inmac, Preis VS. 06102/23684 (Ralf)

2x25 Watt HiFi Hitachi Receiver für DM 150.-, Dieter Lepelmann (069/853259)

Suche

Suche Atari SM124 Monochromonitor, Preis VS. Oli (06103/29990)

Die Gewinner des Preisrätsels aus ST VISION Nr. 6

Zu gewinnen gab es das neue GFA-Basic 3.0 sowie 10 Diskettentaschen der Firma GFA. Herrn Krippner der Firma GFA möchten wir an dieser Stelle für seine Unterstützung danken.

Und hier sind sie nun, die Gewinner! Der Hauptpreis ging an:

Peter Kirchgeßner
Laubenheimerstr.78
6500 Mainz 1

Die Trostpreise gingen an:

Volker Simon, Gießen, Bernhard Milzetti, Langen, Christian Pedersen, Dreieich, Thomas Reihman, Frankfurt, Dieter Jost, Neu-Isenburg, M. Jakob, Marburg, Paul Lukas, Darmstadt, Siegfried Hahn, S. Schulte, Höchst, E. Lorenz, Wiesbaden.

BTX

Eine kurze Einführung

Btx - ein Mitteilungsdienst der Deutschen Bundespost mit rund 3500 Anbietern, die dem Kunden Informationen und Dienstleistungen zur Verfügung stellen.

Zur Teilnahme am Btx-Betrieb gibt es mehrere Möglichkeiten. So bietet die Post das Multi-Tel an. Dies ist ein spezielles Bildschirmtexttelefon für den Btx-Betrieb. Eine weitere Möglichkeit ist ein Btx-Decoder zum Anschluß an ein normales Fernsehgerät.

Da dies alles recht kostenintensive Möglichkeiten sind, bleibt noch die Alternative den schon vorhandenen Computer zu verwenden. Hierzu ist jedoch ein Softwaredecoder nötig. Dies ist ein Programm, das die grafischen Symbole und Zeichensätze interpretiert und auf dem Bildschirm als Bilder darstellt.

Ein Zugang zu Btx erfolgt über die normale Telefonleitung zum Ortstarif. Die Übertragungsrate beträgt normalerweise 1200/75 Baud, wie sie mit vielen Akustikkopplern erreicht werden kann. In einigen Großstädten ist ein Zugang auch mit 1200/1200 oder 2400/2400 möglich. Dazu ist jedoch ein Modem nötig. Dies kann auch von der Post erstanden werden. Nicht FTZ zugelassene Modems arbeiten zwar mit Btx problemlos, jedoch ist ihr Betrieb verboten.

Unabhänig von der Hard- und Software, muß bei der Post eine Anschlußkennung beantragt werden. Dies kostet einmalig DM 65.- und monatlich DM 8.-. Über sie erfolgt eine Abrechung der Kosten während des Betriebes.

ST VISION wird auf jeden Fall weiterhin am Btx-Ball bleiben!

LAYOUT ST

Platinenvorlagen auf dem Atari ST im Eilverfahren erstellt - Nach Digital ST ein weiterer Leckerbissen für Elektronikfreaks

Nachdem in der letzten ST Vision der Logiksimulator ST Digital aus dem Heim Verlag vorgestellt wurde, mußte sich diesmal mit Layout ST ein Programm zur Erstellung von Platinenvorlagen auf unserem Softwareprüfstand auseinandernehmen lassen. Eine Demoversion von Layout ST ist übrigens in unserer PD-Library vorhanden (siehe auch Seite 61 - 69)!

Noch kein Handbuch...

Layout ST ist so brandneu, daß noch nicht einmal das Handbuch fertig gedruckt ist: wir begnügten uns aber dafür gerne mit einem Vorabausdruck auf Endlospapier. Das endgültige Handbuch soll jedoch, wie uns versichert wurde, bis zum Erscheinungsdatum dieser ST Vision verfügbar sein. Mit 15 Seiten verdient es weder Lob noch Tadel: Zu jeder Funktion wird das Wichtigste erklärt, doch hätte der Text manchmal etwas ausführlicher sein können. Da jedoch das Programm sehr einfach zu bedienen ist, wird das Handbuch sowieso schnell im Regal verschwinden.

Auch auf kleinen Systemen

Als Hardwarevoraussetzung genügen bereits ein Atari ST mit mindestens 320 KB freiem Speicher, ein einseitiges Diskettenlaufwerk und ein Monochrommonitor. Zum Ausdrucken des erstellten Layouts wird ein Epson-kompatibler Drucker benötigt. Zum Testen verwendeten wir einen Epson FX-80, der einwandfrei von Layout ST angesteuert wurde.

Um gleich Verwechslungen vorzubeugen: Layout ST stammt nicht (wie Digital ST) aus dem Heim Verlag, sondern aus dem Hause Petersen- Software. Es wird auf einer nicht kopiergeschützten einseitig beschriebenen Diskette komplett mit Handbuch ausgeliefert.

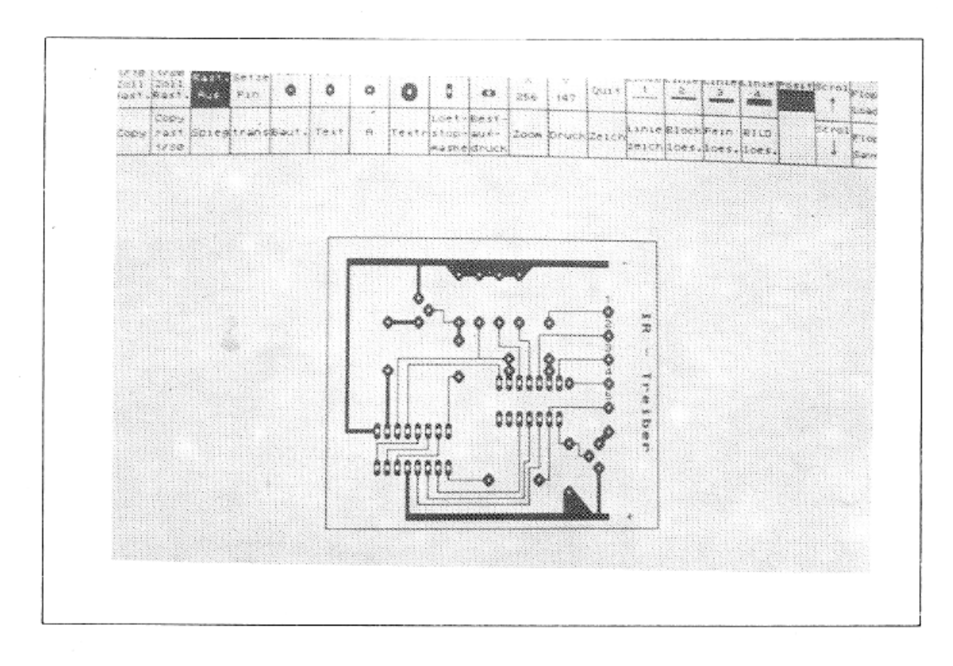
Gut geschützt

Um Raubkopierern dennoch das Leben schwer zu machen, muß beim Programmstart das 8-stellige Kennwort eingegeben werden. Dies scheint einen gelungenen Kompromiß zwischen Kopierschutz und unkomplizierter Anwendung darzustellen.

Nach der Eingabe des Kennwortes präsentiert sich dem Anwender nicht die gewohnte GEM-Menüleiste, sondern der in Bild 1 erkennbare Menübalken. So wird schnellster Zugriff auf alle Funktionen gewährleistet. Als Ausdruck liefert Layout ST eine Vorlage im Maßstab 1:1, was für den Hobbyanwender sicherlich das praktischste ist.

Die Platine darf bis zu 20x20 cm groß sein, wobei die Genauigkeit 1/80" (0,32 mm) beträgt. Professionelle Platinenhersteller werden diese Vorlagen jedoch vermutlich nur mit einem müden Lächeln betrachten, da die Genauigkeit bei diesem Maßstab gerade an der Grenze des Akzeptablen liegt: Maßstäbe von 2:1 oder 4:1 sind hier üblich.

Durch Neudefinition der Bauteile ist es möglich, diese Maßstäbe selbst zu definieren. Bei 2:1 beträgt die maximale Platinengröße 10x10 cm, bei 4:1 können noch Platinen von 5x5 cm bearbeitet werden. Die Genauigkeit bleibt im Ausdruck konstant bei 0,32 mm, jedoch



muß dieser Wert in Relation zum Maßstab gesetzt werden, wodurch man im Maßstab 4:1 auf eine Genauigkeit von 0,08 mm kommt, was für professionelle Feinstleitertechnik durchaus geeignet ist.

Eigener Filmserivce für Notfälle

Sollte dem Anwender kein ausreichend exakter Drucker zur Verfügung stehen, kann der vom Autor zur Verfügung gestellte Filmservice in Anspruch genommen werden, wodurch man vom eigens erstellten Layout reprofähige Filmvorlagen bekommt.

Es geht aber auch anders!

Dem Selfmade Man wird empfohlen, im Copycenter um die Ecke eine Klarsichtkopie vom Ausdruck anzufertigen und als Belichtungsvorlage zu verwenden. Diese Technik habe ich bereits zu Zeiten von Platine ST verwendet. Sie hat zwar ihre Tücken, was mögliche Verzerrungen von ausgeleierten Kopierwerken oder Verschiebungen der zwei übereinanderzulegenden Folien angeht, ist aber trotzdem empfehlenswert.

Also liegt die Hauptanwendung für Layout ST überwiegend im privaten Bereich, und zwar genauer gesagt für den schnellen Entwurf kleiner einseitiger Platinen. Was bisher mit Bleistift und Abreibesymbolen entstand, soll nun auf Bruchteile vom Millimeter genau aus dem Drucker kommen, und zwar möglichst "Dalli Dalli", also schnell und unkompliziert.

Und genau hier liegt auch die Stärke von Layout ST: Die bei objektorientierten Layoutsystemen obligatorische Definition von Bauteileliste und Verbindungsliste entfällt, weil Layout ST pixelorientiert arbeitet. Anders ausgedrückt ist Layout ST ein einfaches Zeichenprogramm, das jedoch auf die Bedürfnisse bei der Entwicklung von Layouts zugeschnitten ist. Der Benutzer fängt sofort zu Zeichnen an und durchläuft genau die gleichen Arbeitsgänge wie be der herkömmlichen Layouterstellung mit Abreibesymbolen.

Viele nützliche Funktionen

So findet man in Layout ST alle Funktionen, die sowohl zur Erstellung des eigentlichen Layouts als auch für die Lötstopmaske und den Bestückungsaufdruck benötigt werden: Leiterbahnen mit genauer x- bzw y-Ausrichtung, verschiedene Lötpunkte, Lupenfunktion, Abruf vordefinierter Symbole (z.B. IC's), Beschriftung in verschiedenen Textgrößen und Textrichtungen, Invertierung ganzer Flächen (für die Lötstopmaske), Raster von 1/10" (2.54 mm-Raster) bis 1/80" und vieles mehr.

Bei mehrseitigen Platinen muß Layout ST zwar nicht unbedingt passen, aber die einzelnen Ebenen (Layer) müssen in getrennten Arbeitsgängen erstellt werden. Lötstopmaske und Bestückungsaufdruck zählen ebenfalls als eigenständige Layer. Eine Umschaltfunktion, die die gleichzeitige Überarbeitung mehrerer Layer erlauben würde, fehlt gänzlich.

Ich persönlich überwand dieses Manko, indem ich mit Multi ST, dem in der letzten ST Vision vorgest ellten Switcher, das Programm Layout ST mehrmals startete und so zwischen den einzelnen Ebenen umschalten konnte. Da selbst der kleinste Atari mit 512KB nach dem Laden von Layout ST noch fast 200 KB frei hat, wäre hier sicher leicht Abhilfe zu schaffen!

Ein anderes Manko ist jedoch viel gravierender: eine UNDO Funktion fehlt! Jeder Ausrutscher kann ganze Bereiche des Layouts unwiderruflich vernichten. Hier gilt ebenfalls: Am Speicher kann's nicht gelegen haben. Abhilfe tut hier dringend Not, zumal sie so einfach wäre (vor jedem Befehl den Bildschirmspeicher kopieren, bei UNDO wieder zurückkopieren).

Forciert wird dieser Fehler durch eher kleinere Unschönheiten wie diese, daß man beim Menüpunkt "Bauteile" den Mausknopf nur ganz kurz anklicken darf, weil sonst ein Bauteil mitten auf dem Layout erscheint. Ohne UNDO wirkt sich dies natürlich verheerend aus!

Für kleinere Arbeiten empfehlenswert

Trotz dieser Probleme stellt Layout ST eine ungemeine Arbeitserleichterung für diejenigen dar, die ein unkompliziertes Layout schneller und sauberer als mit Abreibesymbolen erstellen wollen. Für große Projekte eignet sich die pixelorientierte Arbeitsweise weniger, aber wer will schon ganze Kuchenblechplatinen auf einem Atari entwickeln?

Interessant: Das geplante Update

Der für Ende des Jahres angekündigte Autorouter, der voll zu Layout ST kompatibel sein wird, könnte Layout ST zum professionellen Layoutsystem aufwerten. ST Vision wird am Ball bleiben und so schnell wie möglich über diesen Autorouter berichten. Lassen wir uns also überraschen!

Layout ST ist für DM 149.- bei der Firma Wischolek - Computertechnik in 4250 Bottrop 2, erhältlich.

Die auch in unserer PD-Library angebotene Demoversion kann hier zum Preis von 15,-- DM angefordert werden.

pky

ST VISION SHOP

Abokarte

Die Bestellung kann ich innerhalb von einer Woche beim ST VISION-Leserservice schriftlich widerrufen. Zur Wahrung der Frist genügt die rechtzeitige Absendung.

Das Abonnement hat folgende Vorteile:

- Der Abonnoment wird automatisch Mitglied im gleichnahmigen Userclub.
- Er kann die mehrmals im Jahr stattfindenden Usertreffen kostenlos besuchen.
- Er bekommt sämtliche Disketten der PD-Library zu vergünstigten Preisen.
- Er bekommt alle Hefte bequem sofort nach Erscheinen frei Haus - ohne Mehrkosten!

Hiermit abonniere ich ST VISION ab der nächsten Ausgabe für mindestens ein Jahr zum jährlichen Vorzugspreis von 30DM. Das Abonnement verlängert sich automatisch um ein Jahr, wenn es nicht bis 6 Wochen vor Ablauf gekündigt wird.

Name, Vorname

Straße, Hausnummer

PLZ, Wohnort

Gewünschter Zahlungsweg (bitte ankreuzen)

( )Bargeldlos durch Bankeinzug

[| |][| |][|]

BANKLEITZAHL KONTONUMMER

Geldinstitut

( ) Betrag ist überwiesen / V-Scheck liegt bei

Datum Unterschrift

Bitte ausreichend frankieren!

Antwort-Postkarte

ST VISION

Leser-Service

Postfach 1651

D-6070 Langen

PD-Library

Die Disketten unserer Library sind zu 99% doppelseitig bespielt. Dies dient Ihrem Vorteil, denn dadurch bekommen Sie zum günstigen Preis möglichst viel neue Software.

Zur Bestellung schneiden Sie Bitte die nebenstehende Karte aus und kleben sie auf eine Postkarte.
Diese senden Sie an:
ST VISION
Stichwort PD-Bestellung
Postfach 1651
D-6070 Langen
Vergessen Sie auch nicht, Ihren vollständigen Absender anzugeben.

Haben Sie nur ein einseitiges Laufwerk, so geben Sie bitte getrennt die Programme an, die Sie auf einer einseitigen Disketten benötigen. Durch den erhöhten Arbeitsaufwand beträgt der Preis DM 12.- pro Disk.

Hiermit bestelle ich aus der ST VISION-LIBRARY folgende PD-Disketten:

Disk-Nr.: ___ | ___ | ___ | ___ | ___ | ___ | ___ |

Zum Stückpreis von (bitte ankreuzen):
( ) 8 DM für Mitglieder
( ) 10 DM für Nichtmitglieder

Stückpreis ______ X Anzahl der Disk's ______ Insgesamt: ______

zzgl. Versandkosten je Bestellung 3.-

Endsumme: ______

Den Betrag habe ich bereits überwiesen, ich habe einen Verrechnungsscheck beigelegt. (Bitte keine Briefmarken!)

Datum Unterschrift

VORSCHAU ST VISION NR. 8

Die Ausgabe November/Dezember erscheint am 31.10.88

Großer Vergleich der gängigen Textverarbeitungssyteme

Auf dem Prüfstand stehen die folgenden Programme:

- 1st Word
- 2nd Word
- Word Up
- Signum
- Le Redacteur
- ST Writer
- Easy Text

Assembler auf dem Vormarsch, wir stellen vor: GFA Assembler, OMIKRON Turbo-Assembler

Große Spielberichte:

Carrier Command, Slaygon sowie weitere Tips zu Tanglewood

Atari-Messe Düsseldorf

Wir berichten über die hier vorgestellten Neuigkeiten für den Atari Computer

Dazu unsere regelmäßigen Seiten über Midi, Omikron Basic, BTX, Assemblerprogrammierung

Impressum

ST VISION erscheint regelmäßig alle zwei Monate. Herausgeber ist der Anwenderclub ST VISION, der eine unabhängige Vereinigung von Atari-Usern darstellt. Orthographische Fehler in den Artikeln sind voll beabsichtigt und besonders für solche Leute gedacht, die gerne die Fehler anderer suchen.
Sämtliche Veröffentlichungen in ST VISION erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes. Warennamen werden ohne Gewährleistung einer freien Verwendung benutzt. Für unaufgefordert eingesandte Manuskripte und Listings keine Haftung. Eine Haftung für die Richtigung der Veröffentlichungen kann trotz sorgfältiger Prüfung der Redaktion nicht übernommen werden. Nachdruck nur mit Genehmigung der Redaktion. Namentlich gekennzeichnete Fremdbeiträge geben nicht in jedem Fall die Meinung der Redaktion wieder.

Herausgeber
ST VISION - international usergroup

Chefredaktion
Kai-Uwe Wahl (kuw)
Michael Schütz (mts)

Softwarevertrieb
Matthias Stumm (mas)

Vertriebsleiter
Jürgen Wehnert (jw)

Autoren dieser Ausgabe
Wolfgang Schmidt (ws)
Ralf Rudolph (pky)
Andreas Ramos (Dänemark)
Peter Bauer (peba)
Ralph Klepper (rk)
Eva B. (eb)
Bernhard Milzetti (bm)

Weitere Mitarbeiter
T. Petz
E. Wels
H. Pusch
Kay Streller

Grafische Gestaltung
Oliver Saalfeld

Anzeigenleitung
Kai-Uwe Wahl

ISSN: 0934 - 6716

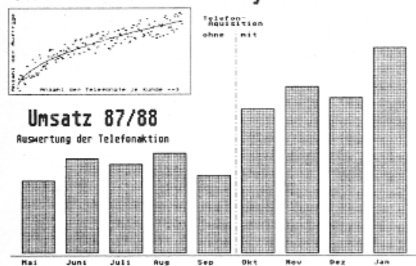
ST VISION ist eine internationale Usergroup für Atari-Anwender. Der Jahresbeitrag beträgt DM 30.- inkl. Versandkosten + MwSt bzw. DM 40.- inkl. Versand Ausland

Unsere Anschrift lautet:
ST VISION
Postfach 1651
6070 Langen
Tel. 06103/1866
BTX 061031866-0001

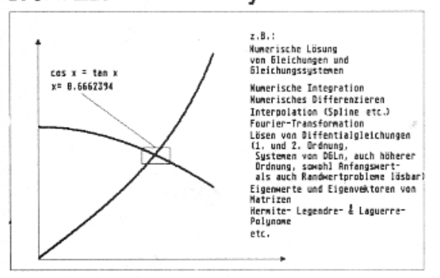
Bankverbindung:
ST VISION, Kai-Uwe Wahl
Bezirkssparkasse Langen
BLZ 505 516 21, Kto. 112 031 650

Postgiroamt Frankfurt
BLZ 500 100 60, Kto. 4520 88-607

STATISTIK - Library



NUMERIK - Library



MIDI - Library

MIDI-Library
für OMIKRON.BASIC

MASKENEDITOR

GUTBUCH-VERSAND ADRESSVERWALTUNG
Name :
Straße :
PLZ/Ort :
Telefon :
Mitglied (J/N) ? Umsatz : DM Datum : Uhrzeit
Datum 2:
Bemerkungen:
Angaben in Ordnung (J/N) oder Adresse löschen (L)?

D.R.A.W.!





OMIKRON.BASIC: das Entwicklungssystem

„Der OMIKRON.BASIC-Interpreter stellt zusammen mit dem Compiler das beste BASIC-Entwicklungssystem für den Atari ST dar."
(ST Computer 9/87, S. 64)

Der Interpreter

Ein umfangreicher Befehlssatz, ein leistungsstarker Editor und höchste Ausführungsgeschwindigkeit kennzeichnen den OMIKRON.Interpreter.

- 19 Stellen Genauigkeit bei allen Funktionen
- Matrizen-Algebra
- Masken-INPUT, SORT-Befehl (mit deutschen Umlauten)
- Strukturiertes Programmieren: rekursive Prozeduren, mehrzeilige Funktionen
- 99 % MBASIC-kompatibel

OMIKRON.BASIC auf Modul **DM 229,–**

* Omikron Basic auf Diskette ist neuer Basic Standard auf Atari ST und liegt seit dem 1.07.88 jedem ST schon ab Werk bei.

Der Compiler

Noch mehr Tempo erreichen Ihre in OMIKRON.BASIC geschriebenen Programme durch unseren OMIKRON.Compiler.
Der Compiler erzeugt ein Assemblerprogramm (echte Maschinensprache, keinen P-Code!), das direkt vom Desktop aus angeklickt werden kann. Auch Accessories können Sie mit dem OMIKRON.Compiler erstellen. Durch CUTLIB werden nur diejenigen Ausführungsroutinen eingebunden, die tatsächlich gebraucht werden; die Folge sind außerordentlich kompakte Programme. Die Ablaufgeschwindigkeit der compilierten Programme ist außergewöhnlich. Der Tester in der mc meinte gar: **„Nur Assembler ist schneller"** (mc 2/87, S. 122)

OMIKRON.Compiler **DM 179,–**
OMIKRON.Compiler für 68881
(vollkompatibel zum normalen Compiler) **DM 229,–**

Die Libraries

Wer professionell Programme entwickelt, möchte nicht jedesmal das Rad neu erfinden. Mit neuen Libraries (Befehls-Bibliotheken) und Hilfsprogrammen macht OMIKRON.BASIC Ihnen das Leben jetzt noch etwas leichter. Libraries erweitern den Befehlssatz: Sie laden einfach die Library, und die neuen Befehle stehen Ihnen sofort zur Verfügung.
Außer den links abgebildeten Programmen gibt's noch eine Robot-Library (ab Mai) und einen **I**ntegrierten **D**ebugger **E**ditor **A**ssembler **L**inker (kurz **IDEAL**), mit dem Sie Maschinenprogramme erstellen und austesten können (ST Magazin 19, Seite 74: „Debugger der Superklasse")

Alle links abgebildeten Programme **je DM 79,–**
IDEAL **DM 99,–**

Gerne schicken wir Ihnen nähere Informationen zu – Anruf genügt!



Programme für Programmierer

OMIKRON.SOFTWARE
Erlachstraße 15 · D-7534 Birkenfeld · ☎ (0 70 82) 53 86

Frankreich: OMIKRON · France, 11 Rue Dérodé, F-51100 Reims
Luxemburg: ELECOMP, 11 Avenue de la gare, L-4131 Esch-Alzette
Niederlande: Terminal Software Publicaties, Postbus 111, NL-5110 Baarle-Nassau
Österreich: Ueberreuter Media, Laudongasse 29, A-1082 Wien
Schweiz: Thali AG, Industriestrasse 6, CH-6285 Hitzkirch